P R O G R A M   M E M O R A N D U M

RAND

Special Operations Forces Aviation Asset Data Model (SOFAADM)

Paul Gaertner

PM-1302-A

March 2002

Prepared for the United States Army

Arroyo Center

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **MAR 2002**	2. REPORT TYPE	3. DATES COVERED **00-00-2002 to 00-00-2002**

4. TITLE AND SUBTITLE **Special Operations Forces Aviation Asset Data Model (SOFAADM)**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Rand Corporation,1776 Main Street,PO Box 2138,Santa Monica,CA,90407-2138**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR’S ACRONYM(S) 11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**110**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

PREFACE

The United States Army special operations forces (SOF) are constantly engaged in operations throughout the world. These forces are deployed in varying numbers from a small insertion team to an entire SOF group. Most recently SOF missions have tended to occur in areas outside major industrialized nations where infrastructures is weak and surface sustainment is difficult. The Special Operations Forces Aviation Asset Data Model (SOFAADM) has been constructed to support the United States Army Special Operations Command (USASOC) in their endeavor to better assist the sustainment and distribution requirements of SOF insurgency teams.

The model is an intra-theater sustainment and distribution model developed in the Extend™ simulation environment which can be used to investigate the general purpose airlift requirements for sustaining deployed SOF units. In addition, the model can be used to examine issues such as SOF basing structures, command and control, unscheduled maintenance, casualty evacuation and routine resupply. The model provides a capability to investigate organic lift versus conventional theater support assets and explore contentions such as shortfalls in intra-theater airlift assets.

This memorandum documents the model and describes some of its potential uses. It is intended for potential users of the model so that they can understand how the model is constructed, its key input and design parameters, as well as various output analysis techniques. This research activity forms part of a larger project providing support for United States Army small and mid-sized operations. It was completed within the RAND Arroyo Center Military Logistics Program. The Arroyo Center is a federally funded research and development center sponsored by the United States Army.

SUMMARY

This memorandum describes the Special Operations Forces Aviation Asset Data Model. The model is an intra-theater sustainment and distribution model that was developed in the Extend™ simulation environment. The model can be configured to a variety of operational structures to investigate the general purpose aviation assets requirements for special operations forces deployments.

An overview of the model is first described and then details are given in subsequent sections. The model is used to characterize both geographic and command and control structures of SOF units deployed in any area of the world. The primary mission is routine supply of food and other consumables needed to sustain the forces using aircraft dedicated to that mission. Then contingency or non-routine missions can be added to assess the impact on the availability of aircraft to meet the demands. Examples of contingency missions would be unscheduled mission planning meetings, emergency resupply, and casualty evacuation. A primary output measure is the percent of time at least one aircraft is available for tasking. This represents the ability to fulfill an unscheduled requirement or mission; this is closely associated with operational risk.

The model also allows the modeler to test different aircraft by inputting their flight characteristics such as airspeed, load capacity, and range limitations. As such it can be used to evaluate new aircraft and compare performances of different aircraft fleet mixes.

The details provided are generally sufficient for a modeler familiar with the basic capabilities of the Extend™ simulation environment to use the model and to develop one’s own scenarios. Detailed descriptions are given of the model’s blocks to include data inputs, what the blocks do, logical connections between blocks, and block and model outputs. It is a model intended for analysts who will first understand its logic and data requirements before using it to investigate a potentially wide range of logistics support issues.

ACKNOWLEDGMENTS

This memorandum documents research undertaken as part of an Australian Defence Science Fellowship to the RAND Arroyo Center in Santa Monica, United States of America, between September 2000 and November 2001.

Although the research described here is my own, it would not have been possible without the assistance of many people and organizations. I am particularly indebted to John Halliday and David Diener who are both members of the RAND Arroyo Center Military Logistics Program. Without their support, research contribution and guidance the development of the model would not have been possible.

Outline

1. Model Overview
2. SOFAADM Description
3. Preparing a SOFAADM Simulation
4. SOFAADM Case Studies
5. Appendices
6. References

Arroyo Center: PM-1302-A RAND

The purpose of this document is to create a depository of development and user's guide information for the Special Operations Forces Aviation Asset Data Model (SOFAADM). The document is divided into the six sections listed in the slide. Section 1 provides a brief overview of the model, the types of mission profiles able to be implemented and the currently used aviation assets. This section concludes by describing some of the model's key assumptions and output variables.

Section 2 provides a descriptive walk through of the model itself. This section is divided into four parts; the *special forces operations base*, the *forward operating base*, the *advanced operating base* and the *SOF team basing structure*.

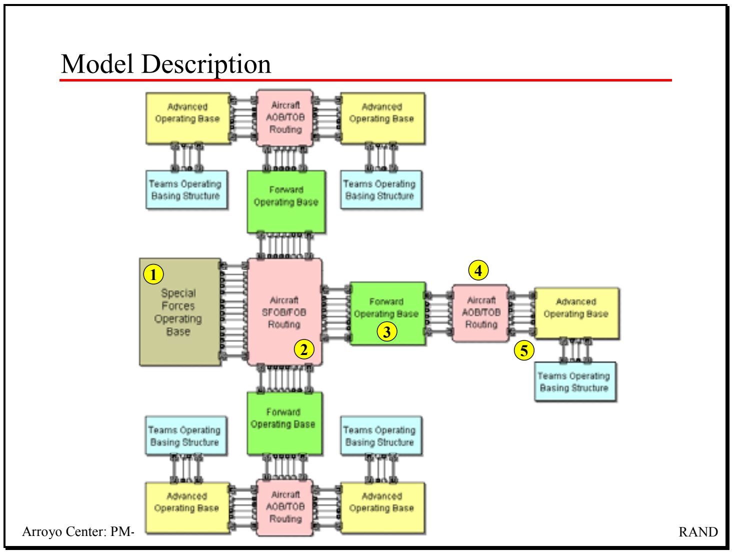
Section 3 describe the procedures for creating tailorable variants of the model and preparing a simulation experiment.

In Section 4, three case studies are presented. These variants of the model are for selected special operations forces groups. This section provides a schematic of the model variant, an indication of the SOF requirements, and a selective portion of output from the simulation.

We also provide appendices and references. These include a list of standard Extend™ simulation modules used in the model and a list of the input and output model parameters.

SECTION ONE

SOFAADM Overview



The structure shown above is but one variant of the special operations forces aviation asset data model (SOFAADM). Developed in the Extend™ simulation environment, the model represents the theater sustainment structure of Special Forces insurgency teams. To date, the model has been used to investigate issues such as SOF basing structures, communication lines, as well as routine and contingency missions.

The SOFAADM structure shown in this slide will be used throughout the briefing to illustrate the functional components of the model. The structure is similar to that of the 5th Special Forces Group and contains a single Special Forces Operating Base (SFOB), three Forward Operating Bases (FOB), five Advanced Operating Bases (AOB), and twenty-five SOF teams (five SOF teams are contained within each of the Teams Operating Bases).

The SFOB module creates and deploys aircraft throughout the structure for the purpose of sustaining the operational force at prescribed levels (rations, ammunition, water etc) or for conducting specified non-routine contingency missions.

Aircraft that depart the SFOB at (1) are routed to a FOB through the *Aircraft SFOB-FOB Routing* block at (2). This block takes the mission profile information imbedded in the parameters associated with each aircraft and then directs the aircraft to its assigned FOB at (3).

Operating in somewhat isolation to each other, each FOB performs two main tasks: (1) it executes the mission assigned to aircraft arriving from the SFOB, and (2) it deploys aircraft to sustain the echelons below it, i.e., the advanced operating bases and the special operating force teams.

Aircraft that depart the SFOB for the FOB are assigned one of two mission profiles, a routine supply profile or a contingency mission profile. A routine supply profile instructs the aircraft to unload a prescribed amount of supplies into the FOB Combat Service Support (CSS) depot. The FOB services an aircraft with a contingency non-routine mission profile separately. Each FOB contains a contingency mission profile manager that routes these aircraft through the base and monitors the execution of their contingency mission. Contingency missions include casualty evacuations, emergency resupplies, and combat planning missions.

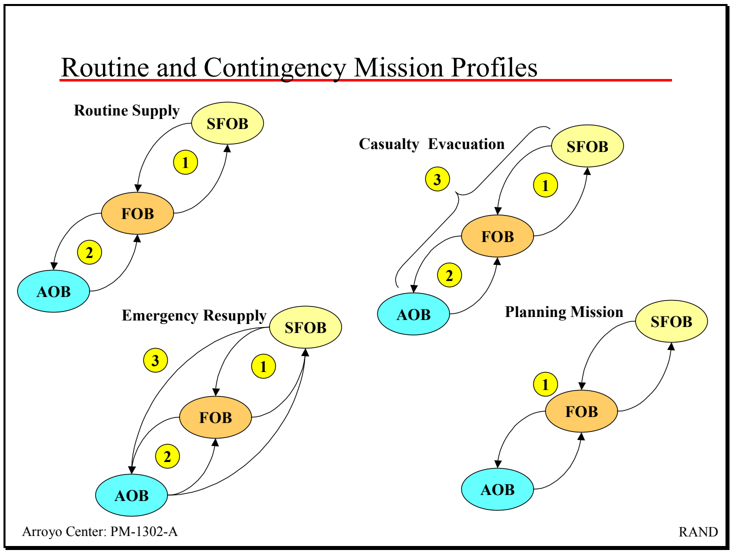
Aircraft that are dispatched from the FOB to the lower echelons operate in a similar manner to that of aircraft departing the SFOB for a FOB, that is, they are tasked with either a routine supply mission or a contingency mission. Aircraft deploying from the FOB to either an AOB or SOF team are routed through the *Aircraft AOB/TOB Routing* block at (4) and (5).

Aircraft arriving at an AOB perform similar tasks as for aircraft arriving at the FOB. Specifically, depending on their mission profile, they either unload supplies into the depot, or they conduct their assigned contingency mission.

At the lowest level are the SOF teams. Each AOB can administer up to five teams. The *Aircraft AOB/TOB Routing* block assists in directing the aircraft through the AOB to each of the teams. As with the AOB and FOB, aircraft arriving at each of the teams either deposit supplies into the depot or conduct an assigned contingency mission.

Depending on the specific implementation of the model and the objective of the experiment being conducted, a number of key performance parameters can be identified. The most commonly quoted output variable is the *percentage of time at least one aircraft is available for tasking*. Other important output variables include *the average number of*

aircraft awaiting missions, *average and maximum aircraft idle times*, the *relative frequency of contingency missions* and the *aircraft workload increase above baseline for contingency missions*.



The generic SOFAADM model allows for two types of mission profiles, routine mission profiles and non-routine contingency mission profiles. Routine mission profiles are the basic function of the model, that is, the supply of rations and ammunition to the forces in the field. The non-routine contingency missions can take one of three forms, *casualty evacuation*, *emergency resupply*, and *mission planning*. This slide briefly describes these mission profiles.

ROUTINE SUPPLY

The routine *supply mission* profile is the basic and most critical mission in the model. This type of mission is used to determine the baseline aviation asset fleet for each experiment. There are two types of routine supply missions, the resupply of an FOB from the SFOB and the resupply of an AOB from a FOB.

NON-ROUTINE CASUALTY EVACUATION

Casualty evacuations are the highest priority contingency mission. There are three types of evacuation missions, an evacuation from the FOB to the SFOB, an evacuation from the AOB to the FOB, and an evacuation from the AOB to the SFOB. This third type is a combination of the first two, that is, aircraft are tasked from both the SFOB and FOB to perform a collaborative mission. Here the aircraft tasked from the FOB retrieves

the casualty from the AOB and on returning to the FOB transfers the casualty to the aircraft that originated at the SFOB.

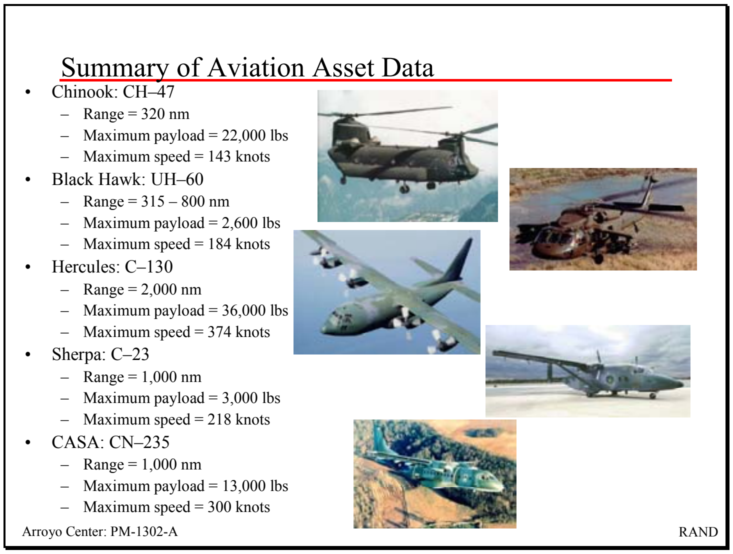
NON-ROUTINE EMERGENCY RESUPPLY

This type of contingency mission is awarded the second highest priority. Three types of emergency resupply missions are available in the model, a resupply of the FOB from the SFOB, a resupply of the AOB from the FOB, and a resupply of the AOB from the SFOB. Unlike casualty evacuations, this third case is not a collaborative mission. An aircraft from the SFOB is flown directly to the AOB.

NON-ROUTINE PLANNING MISSION

This type of mission is assigned the lowest priority. These missions are usually conducted between three levels of the basing structure. For instance, a planning conference scheduled at an FOB will include participants from the SFOB and from one or more of the AOBs administered by that FOB. Likewise, a planning conference scheduled at an AOB will include participants from the FOB and from one or more of the SOF teams. This type of mission is collaborative in that once a planning mission has been scheduled, aircraft are dispatched from respective bases to collect personnel, with the aim of minimizing the time individuals are idle. For example, in the case where a planning conference is scheduled at an FOB, at the time of scheduling aircraft will be dispatched from both the SFOB and the FOB. While the aircraft dispatched by the SFOB transports personnel directly to the meeting, the mission of the FOB dispatched aircraft is to collect personnel from the relevant AOB and transport them to the FOB.

It is important to realize that this slide only illustrates the mission structure between SFOB, FOB, and AOB bases. A similar structure is also present in the model that represents the FOB, AOB, and SOF teams basing structure.



Typically, any aircraft (current or future) can be modeled. However, during any single experiment a maximum of five aircraft types are allowed. The five default aircraft available are the *Chinook* CH-47, the *Black Hawk* UH-60, the *Hercules* C-130, the *Sherpa* C-23, and the *CASA* CN-235.

There are essentially only three parameters required for each aircraft type in the model; they are maximum range, maximum payload, and maximum airspeed. Shown in this slide are the values for the models default aircraft. Readers should take note that the values represented here are merely indicative of the aircraft type. Values may differ dramatically for specific variants of the aircraft.

Although it is relative easy to implement or change an aircraft’s specification, care must be taken. In the current model implementation, all the instances of the aircraft must be changed individually. It is intended to imbed this information into the aircraft parameter set in future updates of the model.

Assumptions and Key Output Variables

Assumptions

- Aircraft maximum flight capacity is 16 hours per 24 hour period.
- Aircraft uplift and unload times are 1 hour in duration.
- Aircraft cost benefit analysis is not performed.
- Next available aircraft corresponds to non-routine missions.
- Frequency of non-routine missions is “scripted random”
- Casualty evacuation, emergency resupply and planning missions are given priorities 1, 2 and 3 respectively.

Key Output Variables

- Average aircraft queue length.
- Average aircraft wait time for scheduling.
- Total number of aircraft departures.
- Percentage of time there was at least one aircraft available.
- Frequency of non-routine contingency missions.
- Percentage increase in aircraft workload over routine supply missions.

Arroyo Center: PM-1302-A RAND

ASSUMPTIONS

- Aircraft maximum flight capacity: The maximum allowable flight hours per 24-hour period for each aircraft is assumed to be 16. Any aircraft that exceeds this limit is immediately stood-down (at its current location) until the required rest period is taken. There are no circumstances that override this constraint.
- Aircraft uplift and unload times: For ease of use, the model assumes that both the uplift and unload times for all aircraft is 1 hour. In future versions of the model this value will be imbedded into the aircraft parameter set, making it easily adjustable between both aircraft type and amount of payload.
- Cost-benefit analysis: As the model presently stands there is no facility for performing cost-benefit analyses between aircraft types. This capability may be added to the model at a later date.
- Corresponding to non-routine missions: Currently, the next available aircraft will respond to a non-routine contingency mission. For example if a casualty evacuation mission is queued from a forward base to an advanced base and a C-130 is at the top of the queue and a UH-60 is second in the queue, the C-130 will undertake the mission, whereas the UH-60 is probably the more

appropriate aircraft. A model revision should consider this and allow individual aircraft to be assigned tasks more appropriate to their capability.

- Frequency of non-routine missions: The frequency of non-routine missions is the responsibility of the user and is controlled via entries in a table created by the user. The model assumes that missions are scheduled on a daily basis. That is, no two missions of the same type can be conducted in the same 24- hour period. A zero entry is assumed to indicate that no mission is scheduled for that 24-hour period.
- Non-routine mission priorities: The model is implemented such that casualty evacuations are given the highest priority, followed by emergency resupply and planning. Routine supply missions are given the lowest priority.

KEY OUTPUT VARIABLES

- Average aircraft queue length: Over the simulation period this parameter represents the average number of aircraft waiting for mission tasking.
- Average aircraft wait time: Over the simulation period this parameter represents the average number of hours a single aircraft spent waiting for mission tasking.
- Total number of departures: This parameter indicates the workload of the aircraft fleet over the simulated time. Workload for individual aircraft can be calculated using this parameter.
- Percentage of time there was at least one aircraft available: This parameter is perhaps the most useful. At the conclusion of a simulation it represents the percentage of time one or more aircraft were available for mission tasking.
- Frequency of non-routine missions: This parameter represents the increase in daily aircraft flights between bases.
- Percentage increase in aircraft workload: This parameter measures the increase aircraft workload over the baseline (routine supply) missions for the aircraft fleet.

SECTION TWO

Special Operations Forces Aviation Asset Data Model (SOFAADM)

Arroyo Center: PM-1302-A RAND

This section describes the Special Operations Forces Aviation Asset Data Model in greater detail. Since it is not possible to describe every module in absolute detail, a working knowledge of the Extend™ simulation environment is assumed.

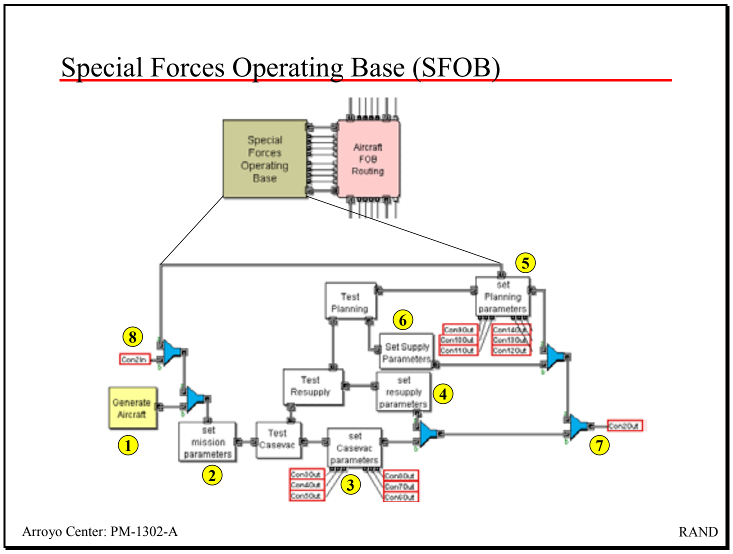
Readers unfamiliar with the modeling environment may wish to first review the material in the appendices. The first appendix provides a brief description of some of the commonly used Extend™ standard modeling terms, the second lists the model input parameter set, and the third lists the model output variable set.

Readers should also understand that in order to run the model both the standard and manufacturing Extend™ software packages are required.

Special Force Operations Base
(SFOB)

Arroyo Center: PM-1302-A RAND

The Special Forces Operating Base (SFOB) is the essential driver for the model. It is here that the aircraft are scheduled for non-routine FOB and AOB level missions and it is from here that routine supply missions throughout all the levels of the model are coordinated.



This slide illustrates the structure of a typical SFOB. There are essentially eight features to the base. While each of these features is described in the following set of slides, an overview is provided here.

1. When a simulation begins the *generate aircraft block* at (1) creates a specified number of simulation items. These items are used to create a number of aircraft and assigns default parameters to them. An aircraft generation block is also present in each forward base which create the aircraft permanently assigned to those locations.

2. The *set mission parameters* block coordinates and schedules aircraft that are assigned to the SFOB. This block is also used to access the *availability* output parameter.

3. The *set casevac parameters* block creates a casualty evacuation mission profile for aircraft that are assigned such a mission. Here the initial mission orders provided to the aircraft are interrogated and expanded to produce a complete mission profile. The output controls connected to the block are used to send information to the appropriate forward base to alert that base to possible collaborative missions.

4. Aircraft arriving at (4) have been scheduled to undertake an *emergency resupply* mission. As in (3) the block uses the mission orders received in the *set mission parameters* block to create the mission

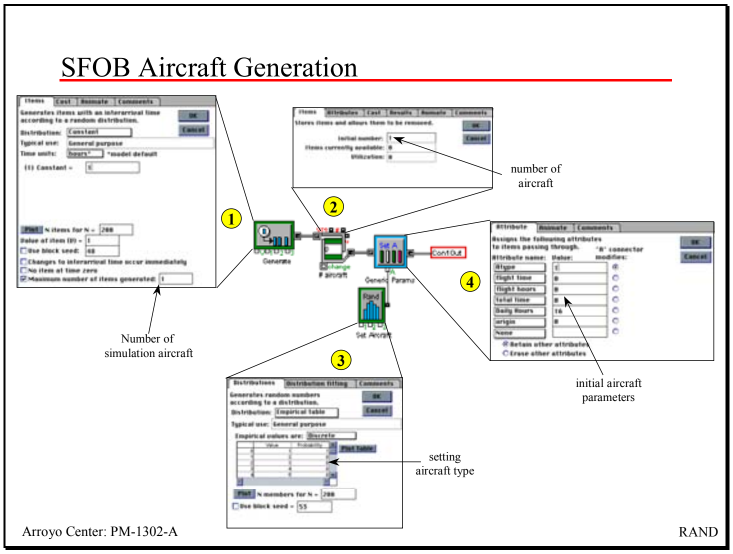
profile. Since these missions do not require any additional work from the forward base, no command information is sent.

5. The *set planning parameters* block at (5) undertakes a similar procedure to that of the *set casevac parameters* block. The only difference is that instead of setting mission parameters for a casualty evacuation, parameters are set for a planning mission. The same information is passed through the controls to the forward bases.

6. The final parameter set block is the *set supply parameters* at (6). Aircraft arriving at this block have been scheduled to undertake a routine supply mission.

7. At (7) all aircraft leave the SFOB and begin their missions.

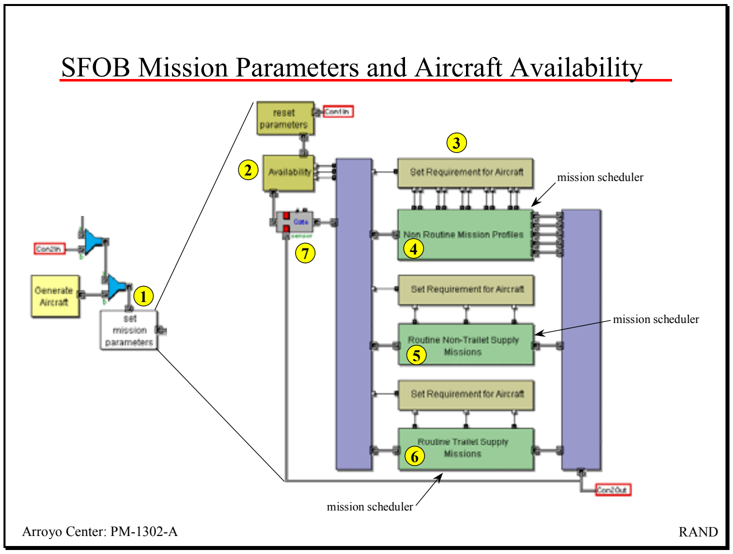
8. Aircraft returning to the SFOB enter at (8). These aircraft are routed back into the *set mission parameters* block where they wait to receive their next mission.



The simulation uses a *generator block* at (1) to create a single simulation item. This is achieved by placing a ‘1’ in the *maximum number of items generated* edit box shown in the insert. The created simulation item is then passed through the *resource block* at (2) where the desired number of aircraft are created. In the example shown in the slide the initial number of aircraft is given as one. It is important to understand that the simulation starts counting from zero. Therefore the ‘1’ placed in the *initial number* edit box will result in the generation of two aircraft.

Each generated aircraft is then assigned a number of parameters, including aircraft type and base of origin. These parameters are shown in the *set attribute* dialog shown in the insert at (4). Since there are several aircraft types available, the *A connector* on the *set attribute* block is used to distinguish between them. This connector is linked to an *input random number* block at (3) and the *radio-button* is set ‘on’ for the variable *Atype* in the attribute dialog. Consequently, the *Atype* (aircraft type) variable will be set to an integer value between ‘1’ and ‘5’ depending on the probability column. The example shown at (3) indicates that 100 percent of the aircraft passing through it will be given an *Atype* value of ‘1’, which corresponds to the Chinook CH-47

helicopter. *Atype* values of 2, 3, 4, and 5 represent aircraft types UH-60, C-130, C-23, and the CASA CN-235 respectively.



Once the desired number of aircraft have been created, they move to the *set mission parameters* block. Aircraft returning to the SFOB after a mission are also routed to this block.

Once an aircraft enters this block all their mission profile parameters, except for those relating to the logged *flight hours* and *total time* in operation, are reset to that of a new aircraft. Essentially, the aircraft is reset to its default values. The values of the number of logged *flight hours* and *total time* in operation parameter are maintained since they are used to determine the operational status of the aircraft and are therefore only reset at the conclusion of each 24-hour period.

The aircraft *availability block* at (2), which is described in detail on the next slide, holds all the aircraft and only releases an aircraft when one has been requested. The *set requirement for aircraft* block at (3) is used to request an aircraft.

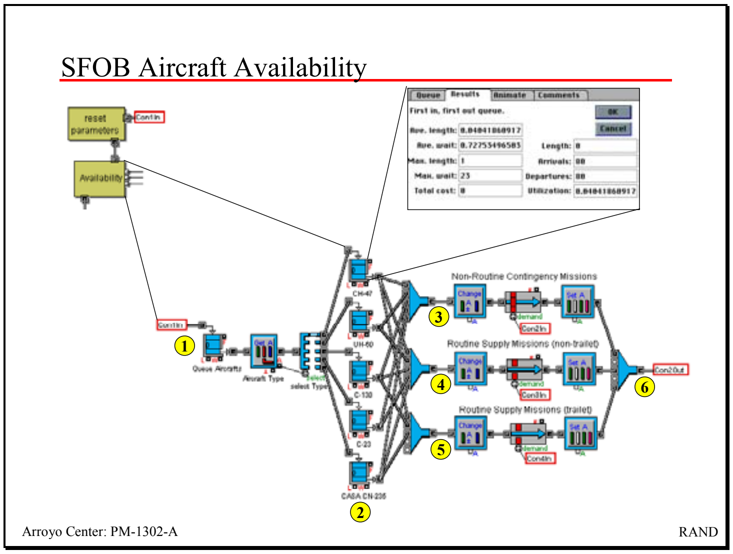
There are two types of missions an aircraft can be assigned. These are routine supply and non-routine contingency. Routine supply missions are categorized as either non-trailet (or per pound amount) or trailet (prescribed amount loaded on a trailer) missions. Routine supply mission types are indicated on the slide at (5) and (6) respectively. Non-routine missions (as discussed earlier) are categorized as either

casualty evacuation, *emergency resupply* or *planning* missions. This mission type is assigned at (4).

Once a mission is scheduled, a command signal is sent through the respective *set requirement for aircraft* block to the *availability* block requesting an aircraft be assigned for that mission. If an aircraft is available for tasking the *availability* block will release it to meet the request.

Once the *availability* block has released an aircraft into the system it proceeds to the respective mission scheduler at (4), (5), or (6). The respective schedulers provide the aircraft with a mission, a destination, and payload parameters. All other parameters required to undertake the mission are set separately in the SFOB parameter section, which is discussed later.

To ensure that aircraft from the *availability* block are routed to the correct mission profile block, only one aircraft is permitted in this section at any time. For example, if two missions are queued simultaneously and therefore request aircraft simultaneously, the second request will not be filled until the first has received its mission orders and left the section. The *gate* block at (7) monitors and coordinates this restriction. Since there are no delays in this section, only simulation time is affected. In real-time it will appear that the two aircraft received their respective orders simultaneously.



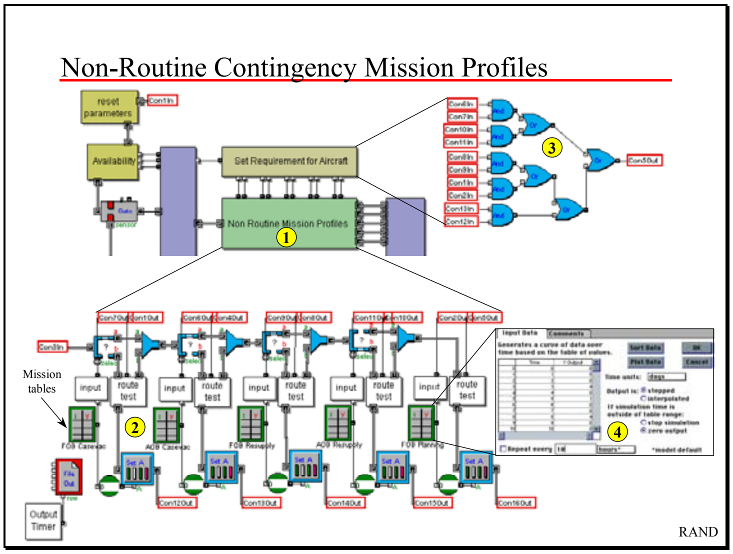
The *availability* block is used to hold aircraft stationed at the SFOB until they are requested to undertake a prescribed mission. The configuration of the *availability* block may differ model to model, but the generic construct is shown in this slide.

Aircraft entering this block at (1) are routed to a series of queues relating to each of the aircraft types at (2). The aircraft types represented here are the Chinook CH-47, Black Hawk UH-60, Hercules C-130, Sherpa C-23 and the CASA CN-235. These queues represent the availability of each of the aircraft types at the SFOB.

The insert illustrates output variables collected within the availability queues. Possible variables are *average queue length*, *average wait*, *maximum queue length*, *maximum wait*, *total cost*, *final queue length*, *total arrivals*, *total departures*, and *utilization*. Although all these variables are useful in analysis, the most important is *utilization*. This variable represents *the percent of time at least one aircraft is available at the SFOB for tasking*. It takes values between 0 and 1; users therefore multiply it by 100 to convert it to a percentage.

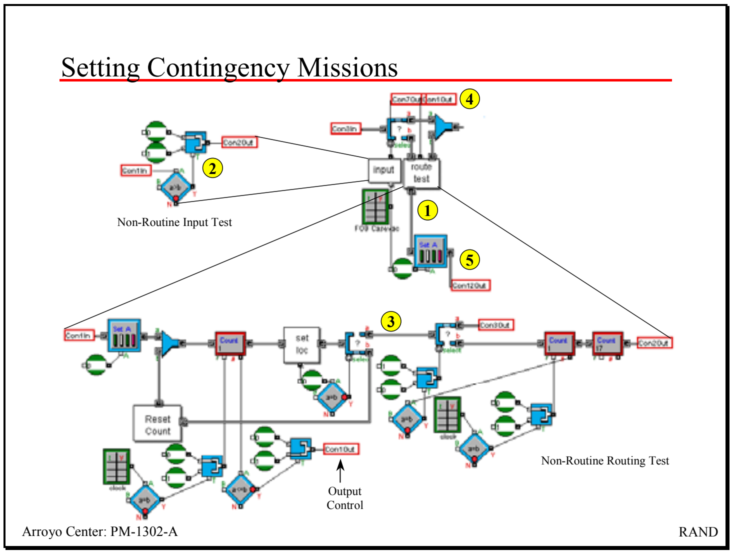
Since each mission type requires a different set of parameters, points (3), (4) and (5) in the slide represent the path taken by each of the mission types.

Once the *demand* block on each of the routes receives a request for an aircraft (through the input connector), the first available aircraft is sent. Note that not all the aircraft types can perform all the missions. Due to weight restrictions the UH-60 and the C-23 are not able to conduct trailet missions. These aircraft are restricted to performing contingency and routine supply (without trailet) missions.



This slide describes the procedure for scheduling non-routine contingency missions. There are two parts to the process. The first is the non-routine mission profiles. Here mission tables are used to control the scheduling of the missions (see (1) and (2)). Each table represents a different mission type (*FOB casualty evacuation*, *AOB casualty evacuation*, *FOB emergency resupply*, *AOB emergency resupply*, and *FOB planning*). The structure of a typical table is shown in the insert at (4). The content of (2) is discussed further on the next slide.

The second part of the process is the *set requirement for aircraft* block. As is seen at (3), this block is a series of linked logical AND and OR nodes. This block receives inputs from the mission profile block (requests for aircraft) and outputs a ‘1’ if an aircraft is requested and a ‘0’ if not, to the *availability* block. If more than one aircraft is requested the output will remain at ‘1’ until all current requests are filled.



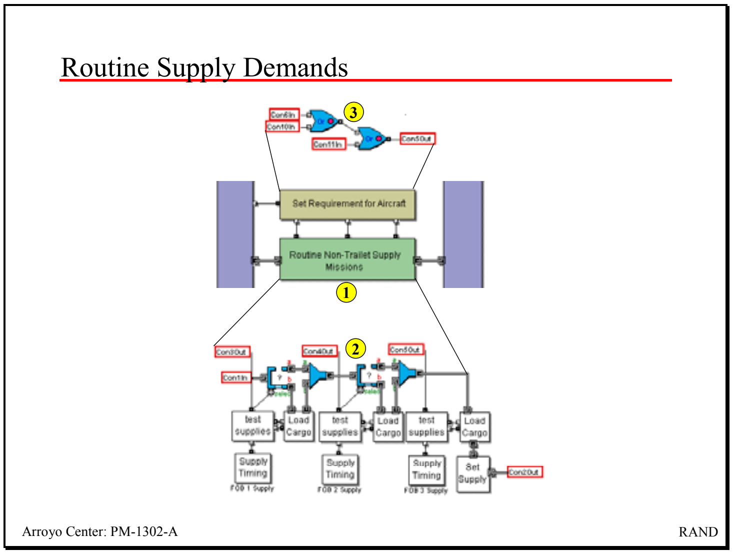
When one of the tables registers that a mission must be undertaken, it sends the mission destination number to both the *input* test block and the *set attribute* block. A tasked mission is indicated by a non-zero value in the mission assignment table. The *input* block at (2) continually tests the input value and outputs a ‘1’ if the value is greater than zero; otherwise it outputs a ‘0’ through the *Con7Out* connector (shown at (4)) to the *set requirement for aircraft* block. This output alone will not cause an aircraft to be tasked. The *route test* block shown in at (3), must also output a ‘1’ value from the *Con1Out* connector for an aircraft to be sent.

The *route test* block is used to ensure that only one aircraft is tasked to conduct any particular mission. It does this by only allowing one aircraft pass through the block to the output at (5) in any 24-hour period. In (3) the *count* blocks containing a value of ‘1’ are reset at the start of every 24-hour period and thus control this process. The counter displaying a value of ‘17’ is used to count the number of aircraft that are tasked this type of mission over the simulated time period. At the start of each 24-hour period, the *Con1Out* connector sends a ‘1’ to the *set requirement for aircraft* block, and continues to send a ‘1’ until an aircraft passes through the block, which changes the value to a ‘0’.

An aircraft can only be requested if the *set requirement for aircraft* block receives a ‘1’ from both the *input* block and the *route test* simultaneously. If either of these is ‘0’ no aircraft is requested.

Once an aircraft has been requested and sent by the *availability* block into the *mission profile* block it will pass sequentially through the contingency missions until the correct mission type is found. Connecting the output from the *input* block to the *select* gate performs this test. A value of ‘1’ will cause the aircraft to enter the *route test* block, whereas a value of ‘0’ will cause the aircraft to move to the next mission type.

As detailed above, the *route test* block is used to ensure that only one aircraft is tasked to conduct the mission. Therefore if an aircraft enters the block and the counter shows ‘0’, that aircraft will be used to conduct the mission and will be passed through the *Con2Out* output to the *set attribute* block and then back into the system. If the counter shows a ‘1’, indicating the mission has already been filled, the aircraft is passed through the *Con3Out* output which passes it back into the sequential testing of mission types.

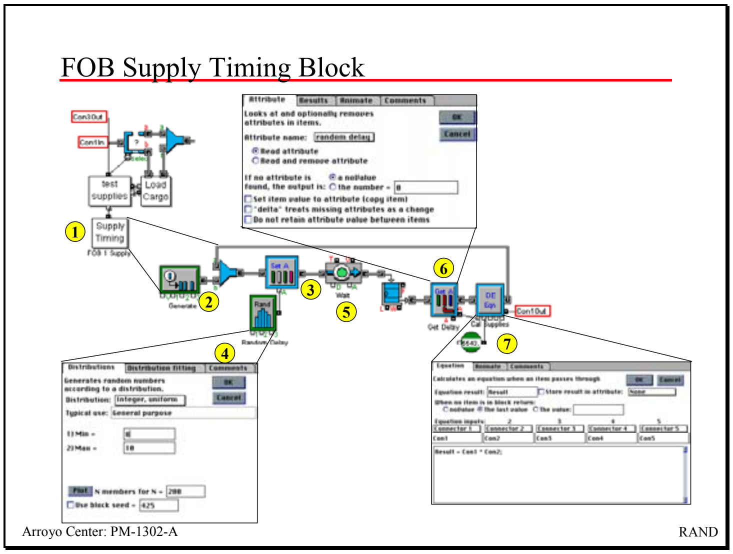


Like the procedure for queuing non-routine aircraft, the procedure for routine aircraft is also a two-step process. The first is the *routine non-trailet supply mission* block at (2), which controls the release of supplies into the system, and therefore requests aircraft to conduct supply missions. The second is the *set requirement for aircraft* block at (3), which is made up of connected logical OR gates that are used to send aircraft requests to the *availability* block. This second step is a simpler version of the *set requirement for aircraft* block that is used for non-routine missions.

When supplies are released from the *supply-timing block,* they move into the *test supplies* block, which registers that supplies are waiting for transport and requests an aircraft through the *set requirement for aircraft* to the *availability* block. Once an aircraft arrives, the *load cargo* block assigns the maximum amount of supplies possible to the aircraft. This amount cannot exceed either the total amount available for transport or the total cargo capacity of the tasked aircraft. Since an aircraft may hold more supplies than the requirement for a single base, if supplies for other bases have been released, the aircraft will visit the *load cargo* blocks for those bases and load supplies up to its remaining capacity. Once an aircraft is fully loaded or no additional supplies are required the aircraft is released from the section. At that

time, if there are additional supplies still requiring transport, a second aircraft will be released from the *availability* block.

The next series of slides discusses the content of the *routine non-trailet supply mission* block in detail.



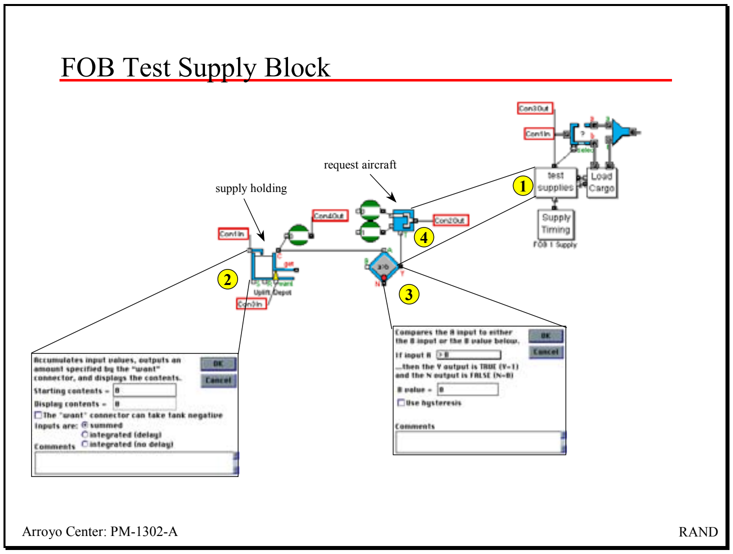
The *supply-timing* block at (1), controls the release of supplies into the system. As was seen on the previous slide, there are *supply-timing* blocks for each of the bases administered by the SFOB. This block functions in isolation to the remainder of the simulation. Its only connection to the simulation is the *Con1Out* connecter, which sends a value representing the amount of supplies to be released to a particular base to the *test supplies* block shown in the next slide.

In the *supply-timing* block, the *generate* block at (2) creates a single simulation item at the start of the each simulation. This simulation item is used to control the release of supplies to a particular base. In the example displayed, this is the ‘FOB 1’ base. Once created, the item passes through the *set attribute* block at (3) which assigns a value to its *random delay* parameter. This parameter represents the number of days that must pass before the next supplies can be released. The insert at (4) illustrates that the delay value is to be an integer value between 8 and 10 inclusive. The *wait* block at (5) carries out the delay.

Once the wait time has been completed the next amount of supplies can be released. Since the aim of the model is to sustain the bases, the amount of supplies released must equal the amount consumed during the wait period. To calculate the required supplies, the daily-consumed

supply for the base is multiplied by the delay time. This calculation is performed at (6) and (7). At (6) the delay time is retrieved from the item and at (7) the amount to be released is calculated. The *constant* block connected to the *equation* block indicates the daily-consumed supplies at the base.

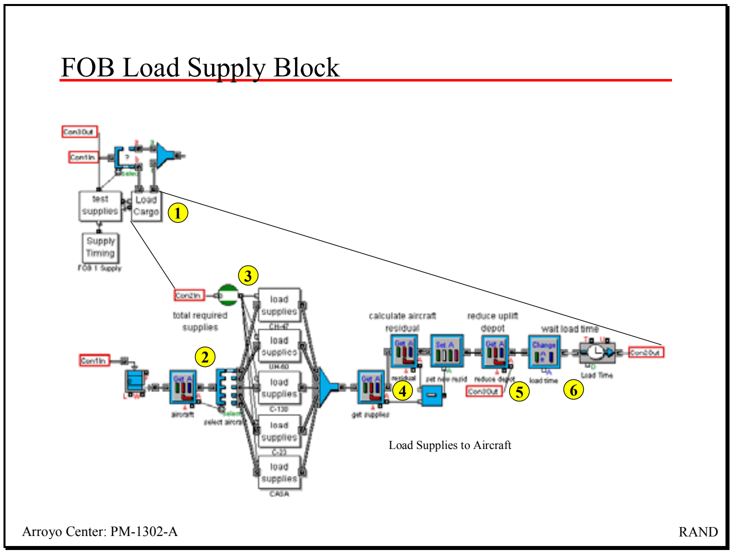
Finally, the output connector releases the supplies to the *test supplies* block on the next slide and the item is returned to the *set attribute* block for the next iteration.



The supplies released from the *supply-timing* block are moved through the *Con1In* connector into the supply holding tank at (2). The insert at (2) indicates that no supplies are currently available for transport. When supplies are available, the non-zero value (representing the amount of supplies) is sent to the decision block at (3) which causes a ‘1’ to be sent through the *Con2Out* connector to the *set requirement for aircraft* block. In turn, this requests an aircraft be sent from the *availability* block to perform the supply mission.

Aircraft that pass through the *load cargo* block load these supplies; this is discussed further on the next slide. Once supplies are loaded, the supply holding tank is decremented by that value through the *Con3In* connector. When the value is returned to zero the *decision* block will switch its output to ‘0’ thereby stopping any further aircraft requests.

The *Con4Out* connector is used to send the amount of supplies waiting for transport to the *load cargo* block.

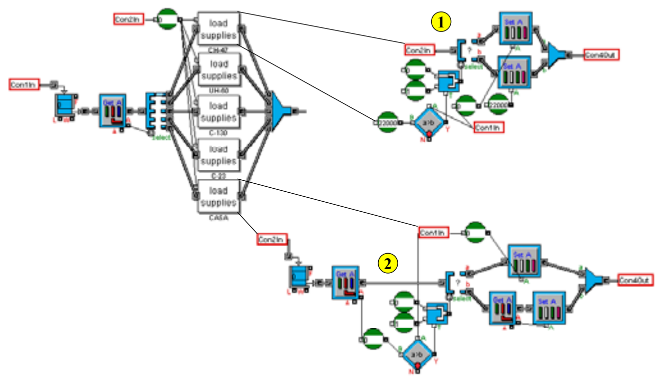


When an aircraft arrives at the *load cargo* block at (1), its aircraft type is determined and it is routed to its respective *load supplies* block at (2), which is discussed on the next slide. The *Con2In* connector at (3) is used to provide each of the *load supplies* blocks with the total amount of supplies required for transport.

Each aircraft is provided with a *residual cargo* parameter, the value of which represents the available carrying capacity of that aircraft type. For the default aircraft types CH-47, UH-60, C-130, C-23 and CASA CN-235 the value of the *residual cargo* parameter is 22,000, 2,640, 36,000, 3,000 and 13,000 pounds respectively. Once an amount of supplies have been loaded onto an aircraft this parameter must be decreased by that same amount. This is performed at (4). At (5) the total amount of the supplies in the supply holding tank (on the previous slide) is also decreased by that amount.

Finally, at (6) a *wait* block is used to represent the load time. Aircraft arriving here are delayed a preset amount of time representing the time needed to load the aircraft.

Loading an Aircraft at the SFOB

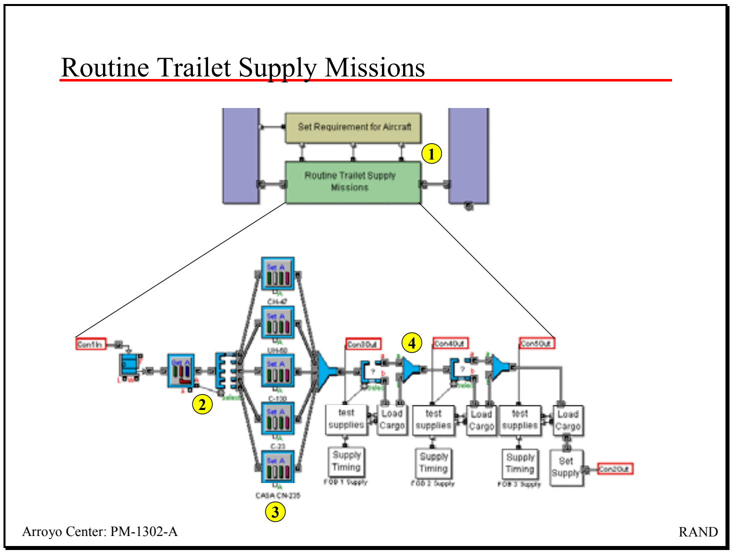


Arroyo Center: PM-1302-A RAND

Depending on state of the aircraft to be loaded, there are two types of loading procedures.

The first is used when an aircraft is empty (the *residual cargo* parameter is set to the maximum capacity of the aircraft) and is therefore being loaded for the first time. In the example, the *Chinook* CH-47 is being loaded. Since this aircraft type has a carrying capacity of 22,000 pounds, that value is the maximum that can be loaded. The *constant* block connected to the *set attribute* block at (1) is set to this value. If the amount of supplies to be loaded is less than or equal to this amount, then the entire amount will be loaded. If not, the maximum capacity is loaded and the remainder must wait for the next aircraft.

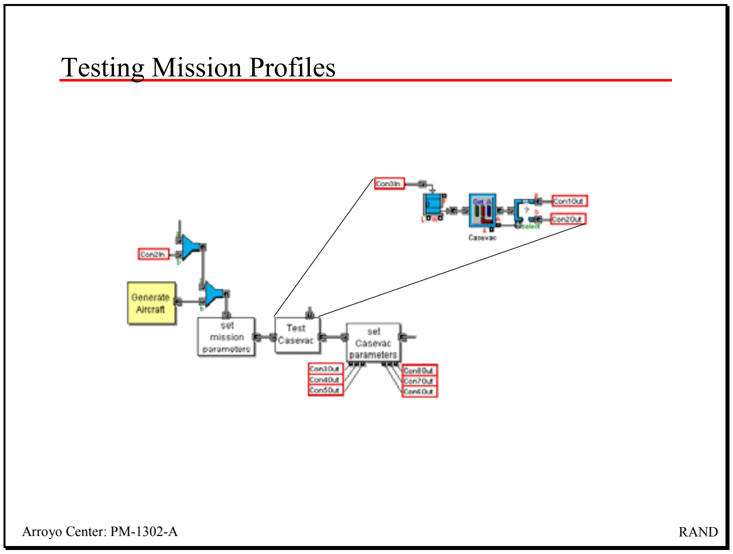
The second procedure is used when an aircraft to be loaded with supplies already has supplies on board. In the example at (2), a CASA CN-235 is being loaded. Here the value of remaining capacity (the *residual cargo*) is retrieved and used as the maximum allowed amount of supplies that can be loaded onto the aircraft.



The procedure for *routine trailet supply missions* at (1) is very similar to that of non-trailet supply missions. The difference is the carrying capacity (*residual cargo*) for each of aircraft type. For instance, the initial carrying capacity of the Chinook CH-47 is given as 22,000 pounds. If a trailet, which is a trailer loaded with a preset amount of supplies is given a weight of 10,000 pounds, then the Chinook will only be capable of transporting 2.

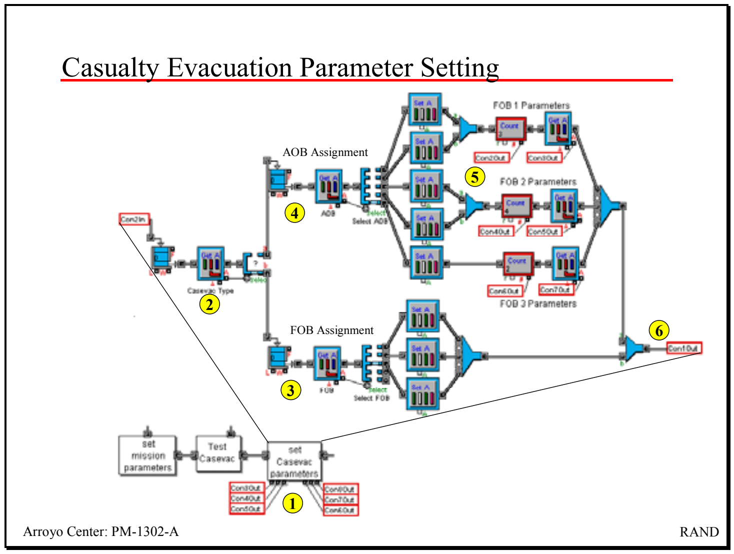
When aircraft arrive at the *routine trailet supply missions* block, they are routed at (2) to their respective *set attribute* blocks at (3) which redefines the total available carrying capacity (the *residual cargo* parameter) to these new values. The loading process at (4) is then the same as for non-trailet supply missions.

This concludes the discussion on the *set mission parameter* block. The next set of slides focuses on the development of mission profiles.



Once an aircraft has received an initial mission profile and/or an amount of supplies to be transported, a number of additional parameters must be set depending on the mission it is assigned.

The first test is whether a casualty evacuation mission has been scheduled. If the test (shown in the slide) is positive, the aircraft is moved to the *set casevac parameters* block, where its parameters are set. The next slide describes the content of the *set casevac parameters* block. If the test is negative the aircraft is moved to the next test and so on until the assigned mission is found.



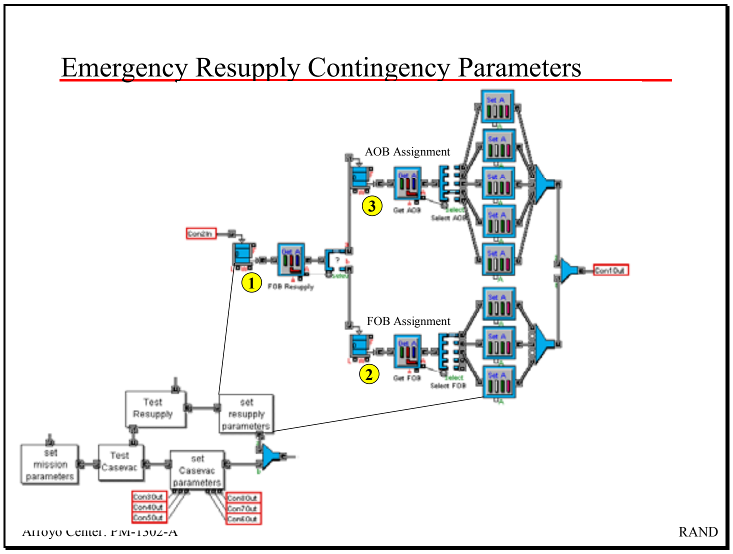
This slide describes the content of the *set casevac parameters* block at (1). A *casualty evacuation* mission can be one of two types, a non-collaborative mission to one of the forward bases or a collaborative mission to one of the advanced bases. The *casevac type get attribute* block at (2) determines which of these mission types is to be undertaken.

If the mission is non-collaborative (to an FOB), the aircraft is routed to (3) where the *select FOB* block is used to route the aircraft to the appropriate *set attribute* block which in turn sets the parameters for that mission. The only parameter required to be set here is the *FOB* parameter indicating to which forward base the mission must be conducted.

If the mission is collaborative (to an AOB), the aircraft is routed to (4) where the *select AOB* block is used to route the aircraft to the appropriate *set attribute* block which in turn sets the parameters for that mission. The example shown in the slide uses five advanced operating bases. If the mission is to either AOB 1 or 2, FOB 1 parameters are set; FOB 2 parameters are set if the mission is to either AOB 3 or 4; and FOB 3 parameters are set if the mission is to AOB 5.

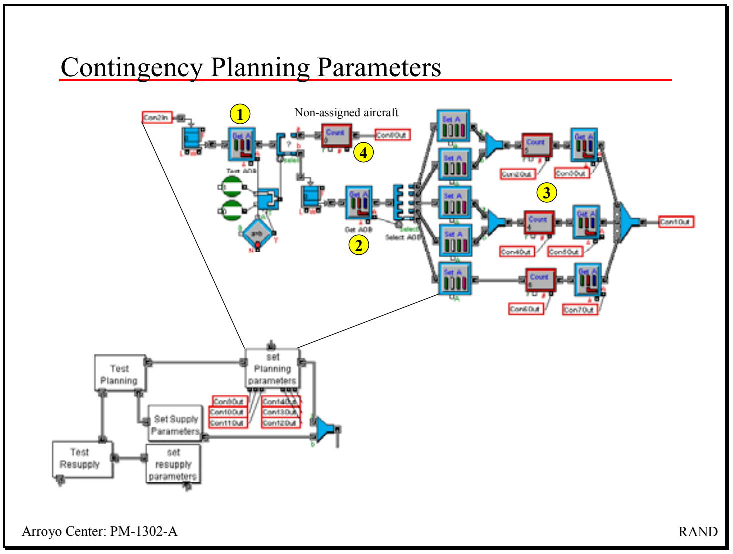
In the case of a collaborative mission the control outputs attached to the *count* and *get attribute* blocks at (5) send information to the

appropriate FOB requesting an aircraft be tasked to the specific AOB to retrieve the casualty. The output from the *count* block sends the mission number to the FOB for housekeeping purposes. The output from the *get attribute* block sends the AOB index number so that the aircraft at the FOB can be directed to the correct base. The aim here is to have the aircraft tasked from the FOB perform its mission and retrieve the casualty before the SFOB scheduled aircraft arrives at the FOB. Finally, the aircraft leave the block at (6).



If the mission type is not a *casualty evacuation*, the second test is for an *emergency resupply*. This mission type is similar to the *casualty evacuation* mission, however, in this case the mission is non-collaborative. Since, from the *emergency resupply* missions can be conducted from the SFOB to either an FOB or an AOB, the FOB resupply *get attribute* block at (1) is used to test which of these two types is to be conducted.

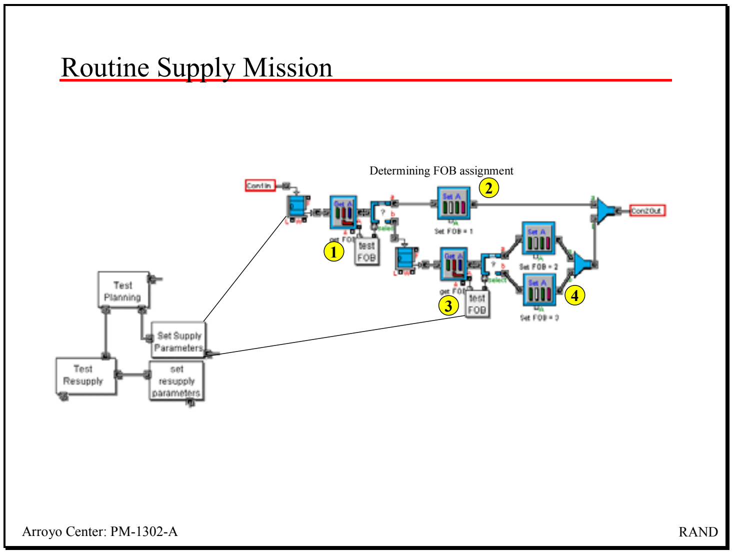
If the mission is to resupply an FOB, the aircraft is routed to (2) and the FOB parameters are assigned. On the other hand, if the mission is to resupply an AOB it will be routed to (3) and the appropriate AOB and FOB parameters are assigned.



The third test and the final non-routine contingency mission test is for a *planning* mission. As is the case for the *casualty evacuation* mission, a *planning* mission is collaborative.

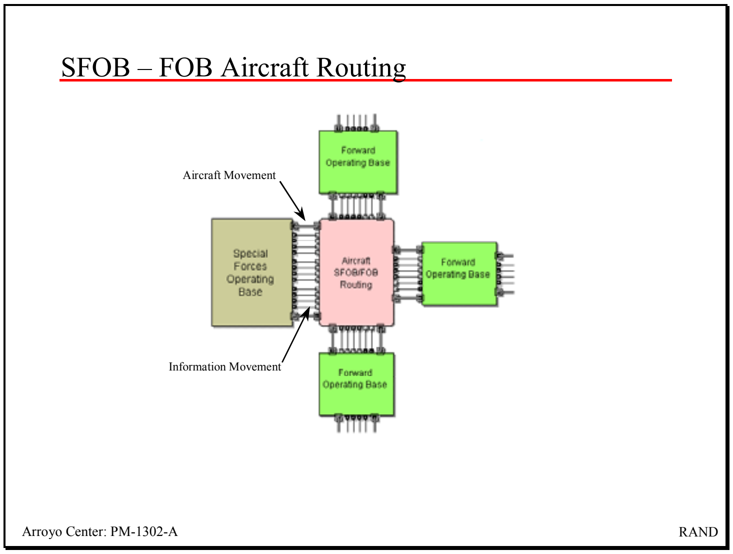
Since this is the final non-routine mission type, arriving aircraft are tested at (1) to determine if a *planning* mission has been scheduled. If the test is negative the aircraft is routed from (4) and is sent back to the *availability* block. This situation is considered an anomaly and rarely occurs. In the event that it does occur, sending the aircraft immediately back to the *availability* block minimizes the effect on the *availability* parameter. The count block at (4) is used to indicate the number of occurrences of this anomaly in the simulation.

If on the other hand the test is positive, which is usually the case, the aircraft is routed to (2) where the appropriate AOB is determined. In this case (as with a *casualty evacuation* collaborative mission) the control outputs attached to the *count* and *get attribute* blocks at (3) send information to the appropriate FOB requesting an aircraft be tasked to the specific AOB to retrieve the personnel for the meeting. The output from the *count* block sends the mission number to the FOB for housekeeping purposes and the output from the *get attribute* block sends the AOB index number so that the aircraft at the FOB can be directed appropriately.

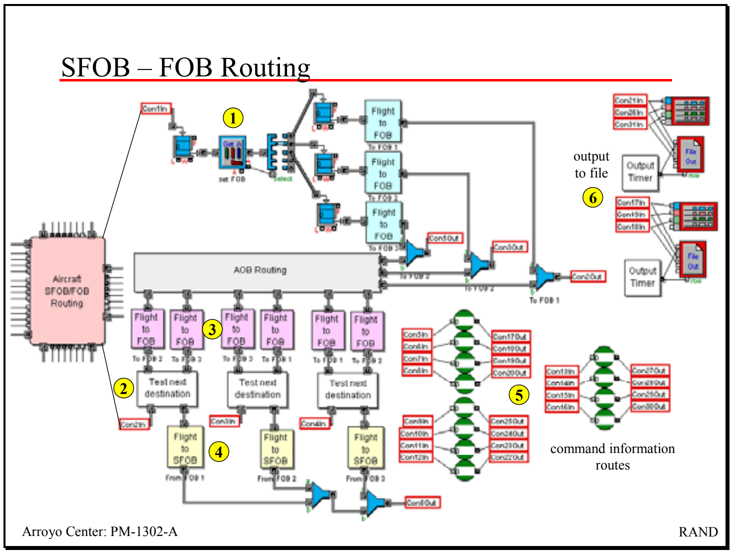


The final SFOB section is the parameter setting for *routine supply* missions. The example shown in the slide is for a three FOB structure. Here the *get attribute* and the test *FOB block* at (1) is used to determine whether the supply mission is to the first forward operating base. If this is the case the aircraft to the *set attribute* block labeled set *'FOB = 1'* at (2) and the appropriate parameters are set. If not the case, a similar test is performed at (3) for the second forward operating base. Again the required parameters are set at (4). Of course if this second test also fails, then the forward operating base must be the third and the parameters are set to reflect this.

The next set of slides describes the routing of aircraft between the SFOB and each of the forward operating bases.



Once an aircraft leaves the SFOB, it enters the *aircraft SFOB/FOB routing* block which is described on the next slide. This block coordinates the movement of aircraft between the SFOB and the forward operating bases. Its function is to track the aircraft as they are processed through to their assigned forward bases and back to the SFOB. In the slide thick links represent the movement of aircraft and the thin links represent the movement of information.



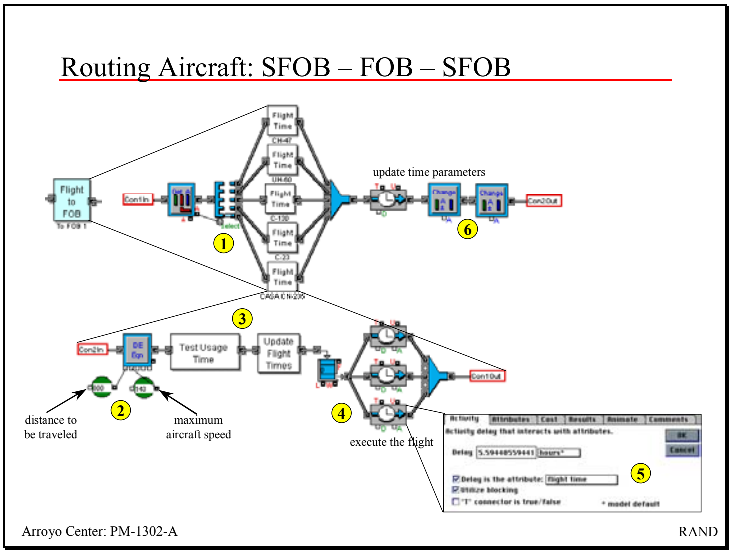
This slide describes the mechanism by which aircraft are routed from the SFOB to several forward bases and back to the SFOB.

Aircraft enter the block from the SFOB through the *Con1In* connector. A test is performed at (1) to determine the first FOB they have been assigned. Aircraft are then routed to that base (shown in blue). These blocks represent the flight time from the SFOB to the particular FOB.

Aircraft returning from an FOB enter the block at one of the inputs (depending on the FOB) at (2). Here the aircraft is tested in the *test next destination* block to determine whether it has completed its entire set of assigned missions or if it is required to visit other bases. In the case where it is required to visit another forward base, the aircraft is flown to that base through one of the pink *flight to FOB* blocks at (3). If no additional forward bases require a visit by the aircraft it is sent back to the SFOB through one of the *flight to SFOB* blocks (shown in yellow) at (4).

The command information links are routed between the SFOB and each FOB at (5). The important thing to note is that you cannot link an input directly to an output, thus a *constant* block (as shown) is used to route the information. Furthermore, since constant blocks increment the input value by the specified constant, these values must be preset to zero.

The output to file and the graphing of results is illustrated at (6). Here the contents of each of the FOB supply depots are displayed and entered into an output file for post-run analyses.

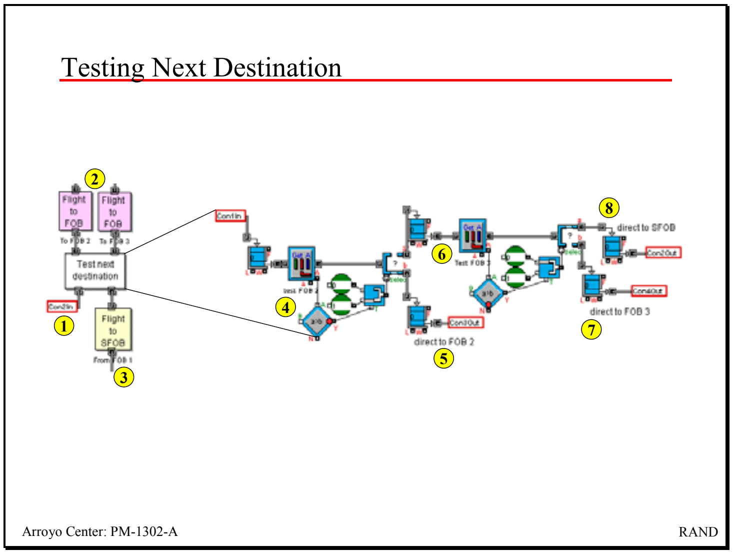


This slide describes a typical *SFOB-FOB flight* block. All flight blocks are similar to the one shown. The only difference between these blocks is the distance that needs to be traveled.

When an aircraft enters this block the aircraft type is determined and the aircraft is routed to the appropriate *flight time* block shown at (1). The content of one of these blocks is shown in the bottom half of the slide.

When an aircraft arrives in this block, the flight time from the SFOB to the FOB is calculated and stored in the aircraft's flight time parameter at (2). Before a flight can be conducted however, it needs to ensure that the total flight hours in the current 24-hour period will not exceeded the maximum allowable hours by undertaking this flight. In the cases where this total will be exceeded the aircraft is stood-down at (3) until the required rest period has been taken. Once the aircraft has passed (3) it moves to (4) where the flight is undertaken. At (4) three delay blocks are provided in parallel indicating that up to three aircraft can be flown on the route simultaneously. The insert at (5) shows that for this particular aircraft the flight time between the SFOB and the FOB is about 5.59 hours.

Once an aircraft has flown the required distance, its housekeeping (time) parameters are updated at (6) and the aircraft moves from the *SFOB-FOB routing* block into the specific forward operating base.

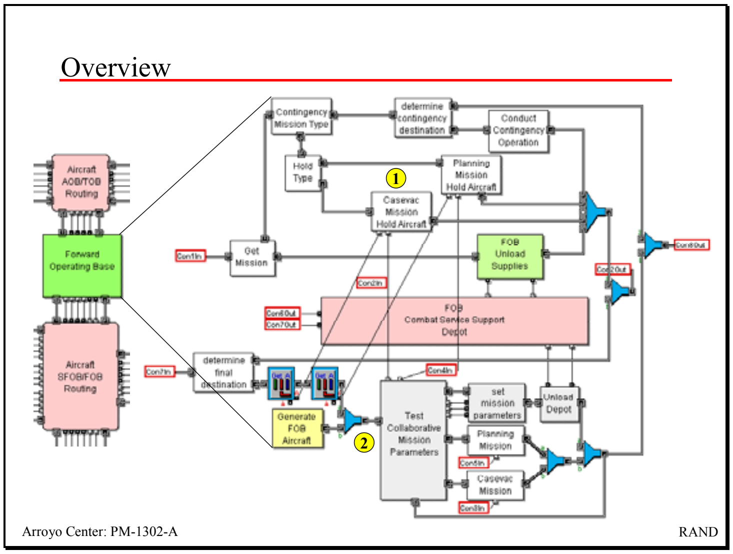


Aircraft returning to the SFOB-FOB routing block after visiting a forward base are tested in the *test next destination* block at (1) to determine if they are required to visit other forward bases before returning to the SFOB. Those that are required to visit other bases are routed to those bases at (2) and those that are not are routed back to the SFOB at (3). Each of the flight blocks (pink and yellow) is similar to the block described on the previous slide.

In this slide it is assumed that the aircraft is returning after visiting FOB 1. Consequently, the possible destinations are FOB 2, FOB 3, and the SFOB. The first test at (4) is for FOB 2. If the aircraft's attribute for FOB 2 is greater than zero the aircraft will be routed to that base at (5). If not, a second test is performed at (6) to determine whether the attribute for FOB 2 is greater than zero; if so, the aircraft will be sent to that base at (7). Finally, if neither is true, the aircraft is assumed to have completed its mission and is returned to the SFOB at (8).

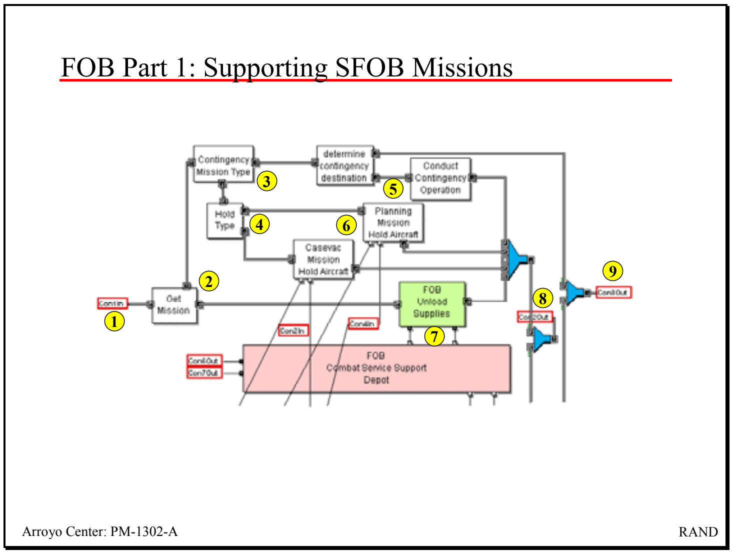
This completes the sub-section relating to the SFOB. The next sub-section describes operations at the forward operating bases.

Forward Operating Base
(FOB)



The model for a forward operating base (FOB) is largely dependent on the structure of the special operating force being implemented. The FOB shown in this slide should be regarded as a typical ‘generic’ forward base.

A forward operating base essentially performs two tasks. Firstly, it processes the arriving aircraft from the SFOB, and secondly, it maintains and coordinates operations between itself, the advanced operating bases and “to some extent” the Special Forces teams. The following set of slides examines the operations of a forward base in detail.

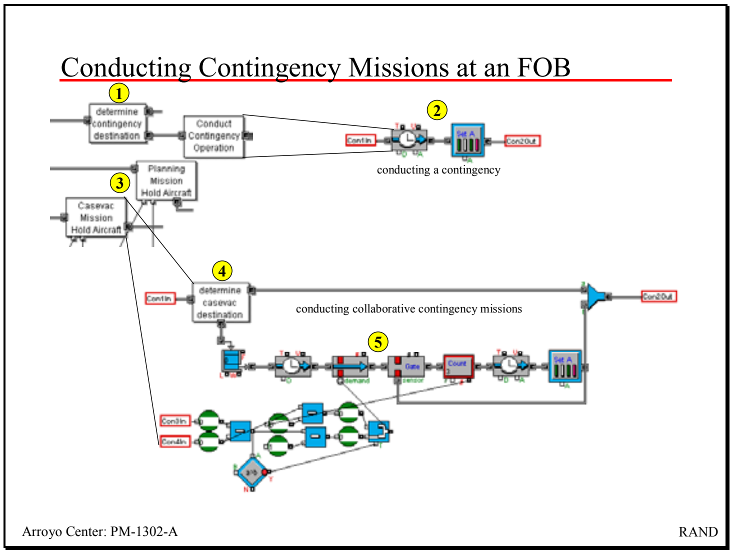


Aircraft arrive at the FOB from the SFOB through the *Con1In* connector at (1). Upon arrival, the *get mission* block at (2) determines whether the mission type is routine or non-routine.

Non-routine missions are routed to (3) where the non-routine mission is further broken down into collaborative and non-collaborative. Casualty evacuation and planning mission types are routed to (4) and then to their respective mission blocks (described in the next slide) at (6). Non-collaborative resupply missions are routed to (5) where a determination is made as to their final destination. Resupply missions that have the FOB as a final destination are sent to *conduct contingency operation* block (described on the next slide). Those that have an AOB as a final destination are routed out of the FOB to the AOB through the output at (9). Once non-routine missions with the FOB as the final destination are complete, the aircraft is routed back to the *SFOB-FOB aircraft routing* block through the output at (8).

If the mission is determined at (2) to be that of a routine supply mission the aircraft is routed to the *FOB unload supplies* block at (7). Here the supplies for this particular base are unloaded into the base combat service support depot and the aircraft is returned to the *SFOB-FOB aircraft routing* block through the output at (8). The unload

supplies procedure and the contents of the depot will be described later in this section.



In non-collaborative *emergency resupply* missions, aircraft either remain on the ground at the FOB for a specified time and then exit the base, or if the final destination is an AOB, immediately exit the base (without landing) for that AOB. The determination between the two is performed at (1).

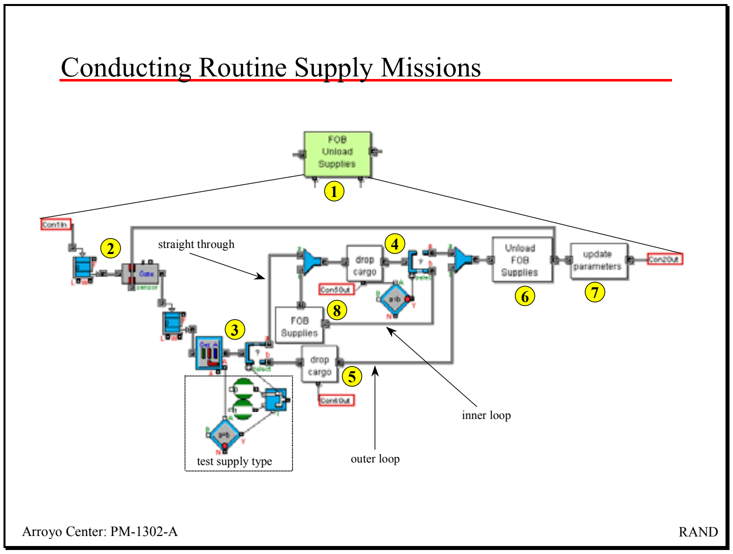
In the case where the FOB is the final destination, the contingency mission is conducted in the *conduct contingency operation* block at (2). Here as mentioned above, the aircraft waits a specified time and then exits the base. The *set attribute* block at (2) is used to reset the contingency mission parameters to indicate that the mission has been completed successfully.

The procedure for *casualty evacuation* and *planning* missions at (3) is a little more complex. In these cases the mission type could be either collaborative or non-collaborative. The determination between these (for casevac missions) is conducted in the *determine casevac destination* block at (4). If the FOB is the final destination, a non-collaborative mission is assumed. In this case the SFOB-tasked aircraft collects the casualty at the FOB and departs. If the final destination is an AOB, the mission is assumed to be collaborative and the aircraft moves into the segments at (5), where the aircraft waits for the casualty to be brought to the FOB from the AOB by an FOB-tasked

aircraft. Upon arrival of that aircraft the casualty is transferred to the originating SFOB aircraft, which then departs the base.

A collaborative mission type operates as follows. Once queued in the SFOB, information sent to the FOB via the *Con3In* connector causes the *demand* block at (5) to be closed stopping any aircraft from passing through it. The information is also used by the FOB to task an aircraft to the appropriate AOB to retrieve the casualty. When this aircraft arrives back at the FOB with the casualty, information is sent into the *Con4In* connector opening the *demand* block allowing the waiting SFOB aircraft to continue its mission. If for some reason the FOB-tasked aircraft arrives back at the FOB before the SFOB aircraft arrives at the block, the SFOB aircraft would pass immediately through the block since the *demand* gate would already be open. The numerical difference between the *Con3In* and *Con4In* connectors is used to determine how many aircraft should be allowed through the system, since a situation could occur where more than one casualty evacuation is operating simultaneously.

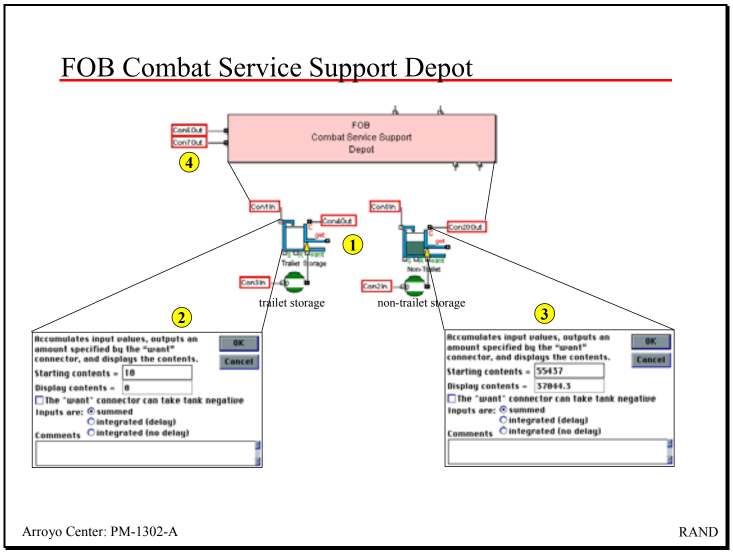
The operation of the *planning* mission block is the similar to that of the casualty evacuation block except that all planning missions are considered collaborative.



The *FOB unload supplies* block at (1) deposits supplies from the aircraft into the FOB Combat Service Support depot, described on the next slide.

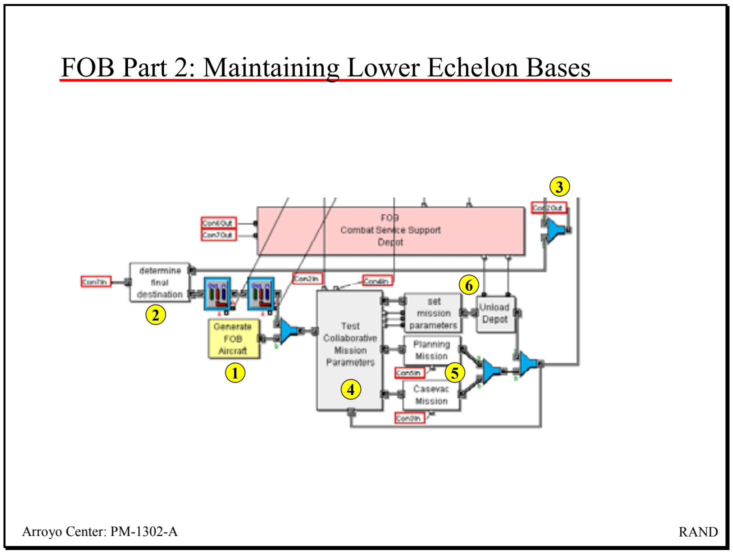
Aircraft arriving at (2) are able to unload two types of supplies, non-trailet and trailet. A test at (3) is conducted to determine which of the types is to be performed. It is not possible for a single aircraft to unload both types in the same visit to the FOB. Non-trailet supplies are routed to (4) where the supplies are passed through the *Con5Out* connector to the depot. Trailet supplies are routed to (5) where the *Con6Out* connector is used to pass the supplies to the depot. Once the aircraft has completed the unload mission at (6), its parameters are reset at (7) and it departs the base.

The *FOB supplies* loop at (8) represents supplies that are consumed at the base. In this case, at preset intervals (usually 8-hours), a negative value (representing the consumed supplies) is sent through the *Con5Out* connector to the depot decreasing the total by that amount.



The FOB Combat Service Support depot contains two holding tanks, one for trailet supplies and the other for non-trailet supplies (shown at (1)). As was mentioned on the previous slide, the depot receives supplies for each holding tank from the *FOB unload supplies* block. The inserts at (2) and (3) illustrate the differences between these supply types. In the trailet case an initial stock of 10 trailers is required, whereas in the non-trailet case an initial stock of over 55,000 pounds is required.

The *Con6Out* and *Con7Out* at (4) are used to transfer the respective total supplies in the depot to the output file and simulation graphical display unit.



The second major function of the FOB is the management and resupply of the lower echelon bases. The structure shown here is similar to the structure of the SFOB. However, at the FOB there is the added difficulty of having to support the SFOB collaborative missions.

As is the case for the SFOB, a *generate aircraft* block at (1) is used to create the initial aircraft and assign initial parameters to them. Since this function is similar to that performed in the SFOB, it will not be described here. However, the aircraft that are in this block are considered property of the FOB and therefore can only be tasked by the FOB. These aircraft cannot travel to the SFOB; they are for FOB and lower echelon operations only.

Aircraft return to the FOB from the lower echelons at (2). Since it is possible for SFOB aircraft to reach the AOB level, the block labeled *determine final destination* is used to ascertain the arriving aircraft’s base of origin and route the aircraft back to that base. Aircraft that originated at the SFOB are routed to (3); all others are routed to the *test collaborative mission block* at (4). The *get attribute* blocks immediately following (2) are used to signal the completion of a collaborative mission.

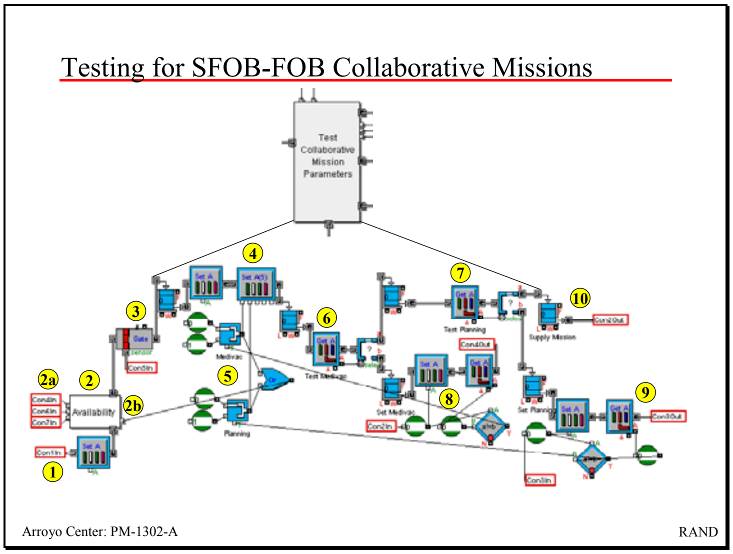
The *test collaborative mission block* at (4) performs two key functions. First, it coordinates SFOB collaborative missions by tasking

FOB aircraft to undertake particular missions. Second, it contains the FOB aircraft *availability* block, which is used to hold the aircraft until a request for their service is received. The *test collaborative mission block* is described further on the slide.

If a collaborative mission is required, the aircraft undertaking that mission is passed through either the *planning mission* block or the *casevac mission* block depending on the mission type at (5) to receive its mission profile. The *Con5In* and *Con8In* connectors provide the SFOB-supplied information required to undertake the planning and casualty evacuation missions respectively.

FOB-scheduled missions are scheduled in the *set mission parameters* block at (6). Although the structure of this block is similar to the *set mission parameters* block in the SFOB, that is, both routine and non-routine missions are available, the FOB scheduling of routine missions includes both AOB and SOF teams (denoted as TOB) instead of FOB and AOB missions.

In some model implementations where the AOB operates its own aircraft, this will not be the case. Instead the SOF teams will be administered by the AOB instead of the FOB.



The aim of the *test collaborative mission* block is to test and set the initial parameters for SFOB-FOB collaborative missions (casualty evacuation and planning). The block is also used to coordinate the FOB aircraft availability mechanism.

Aircraft arrive in this section at (1) and are routed to the *availability* block at (2) where they are held until a request is received through one of the input connectors for their support. Whereas the input connectors at (2a) are used to request aircraft for FOB-related missions, the input connector at (2b) is used to request aircraft for collaborative missions.

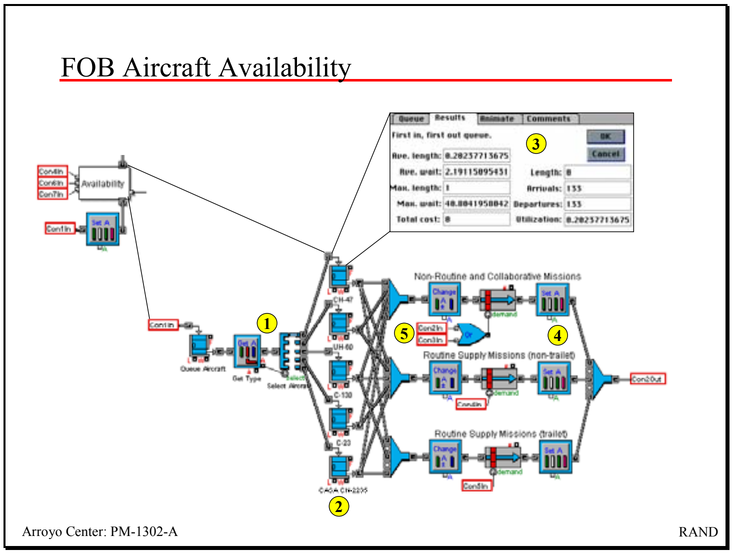
Once leaving the *availability* block, aircraft move through the *gate* block at (3) and officially enter the system. Note that the *gate* block ensures that only one aircraft is entered into the system at a time. This is an essential requirement so that the switching sequence can operate correctly.

The remainder of the *test collaborative mission* block either sets the collaborative mission initial parameters and directs aircraft undertaking those missions to their respective outputs, or directs aircraft to the FOB *set mission parameters* block.

The assigning of collaborative missions to FOB aircraft is controlled at (4) and (5). Here, if either of the switches is set to

‘1’ a control is sent through the OR gate to the *availability* block requesting the release of an aircraft. When an aircraft is released for a collaborative mission, the *set attribute* block attached to the switches will set appropriate parameters so that the model identifies this aircraft as assigned to a SFOB-originated mission and therefore will not assign it to other tasks.

Points (6) and (7) ensure that the aircraft is routed correctly through the system to its set parameter location. At (6) a test is conducted to determine if a collaborative casualty evacuation mission is to be undertaken. If the test is positive, the aircraft is routed to (8) and exits the block via the *Con4Out* control. If the test is negative, the aircraft moves to (7) where a test is performed to determine whether collaborative planning is to be performed. Again if the test is positive, the aircraft is routed out of the block, this time via the *Con3Out* control at (9). If the planning test is negative, the mission is assumed to be an FOB-scheduled mission and the aircraft leaves the system at (10) via the *Con2Out* control.



The *availability* block is used to hold aircraft stationed at the FOB until they are requested to undertake a prescribed mission. Although the configuration of the *availability* block may differ from model to model, the generic construct is shown in this slide.

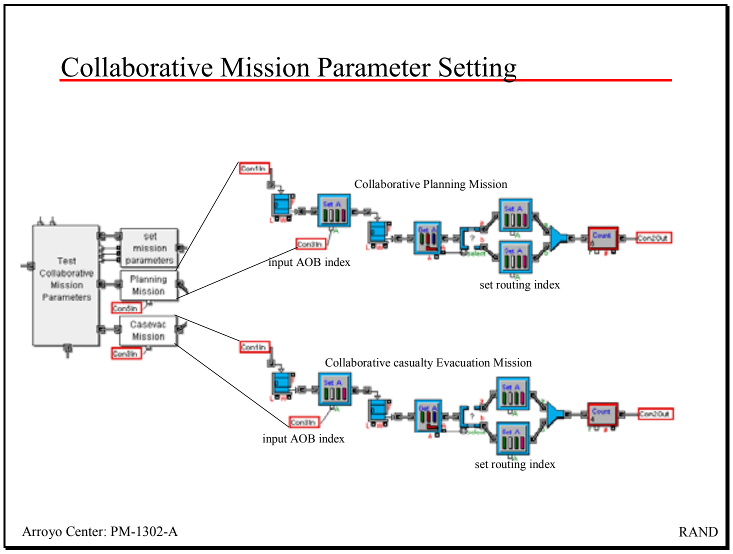
Aircraft entering this block at (1) are routed to a series of queues relating to each of the aircraft types at (2). The aircraft types represented here are the Chinook CH-47, Black Hawk UH-60, Hercules C-130, Sherpa C-23, and the CASA CN-235. These queues represent the availability of each of the aircraft types at the SFOB.

The insert at (3) illustrates the contents of the availability queues. The parameters of the queue block are used as model output variables: *average queue length*, *average wait*, *maximum queue length*, *maximum wait*, *total cost*, *final queue length*, *total arrivals*, *total departures*, and *utilization*. Although all these variables are useful in analysis, the most important is the *utilization*. This value represents *the percent of time at least one aircraft is available at the SFOB for tasking*. This variables takes values between 0 and 1 and is multiplied by 100 to convert it to a percentage.

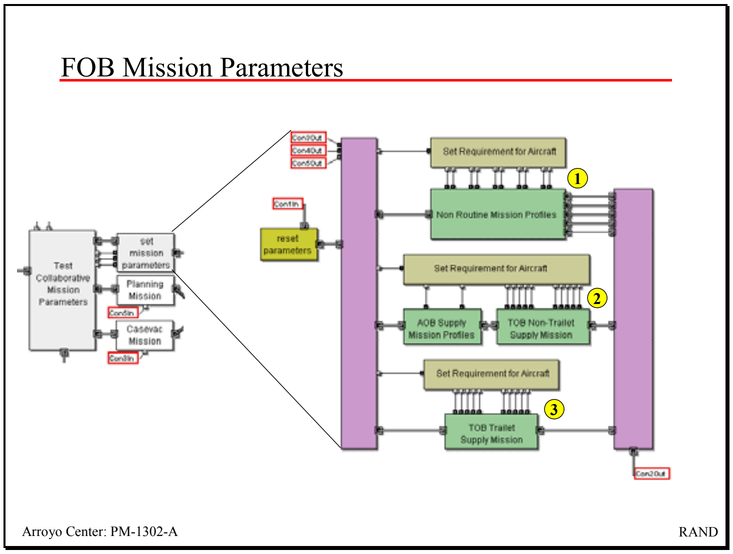
Since each mission type requires a different set of parameters, point (4) illustrates the path taken for each of the mission types. The

two inputs at (5) represent non-routine FOB scheduled missions and SFOB-initiated collaborative missions.

Once the *demand block* on any of the path receives a request for an aircraft (through the input connector), the first available aircraft is tasked. Note that not all aircraft types can perform all the missions. Due to weight restrictions, the UH-60 and the C-23 are not able to conduct trailet missions. These aircraft are restricted to performing contingency and routine supply (without trailet) missions.



This slide demonstrates the final parameter setting blocks for the collaborative casualty evacuation and planning missions. Effectively, the only parameter required to be set comes from the *input AOB index* representing the AOB to which the aircraft must travel. This index is provided as information from the SFOB and is entered through the *Con3In* connectors in each of the collaborative mission blocks.

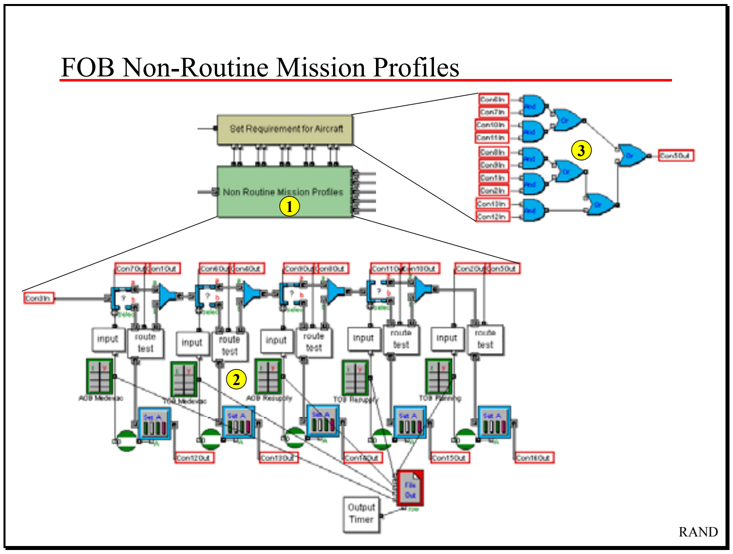


When an aircraft enters this block all its mission profile parameters (except for those relating to the logged *flight hours* and *total time* in operation) are reset to that of a new aircraft. Essentially, the aircraft is reset to its default values. The values of the number of logged *flight hours* and *total time* in operation parameter are maintained since they are used to determine the operational status of the aircraft and are therefore only reset at the conclusion of each 24-hour period.

There are two types of missions to which an aircraft can be assigned, routine supply and non-routine contingency. Routine supply missions are categorized as either non-trailet (or per pound amount) or trailet (prescribed amount loaded on a trailer) missions. Routine supply mission types are indicated at (2) and (3) respectively. This type of mission can be conducted to either of the advanced operating bases or to one of the SOF teams (TOB). Non-routine missions, categorized as either casualty evacuation, emergency resupply or planning missions are assigned at (1).

Once a mission is scheduled, a request is sent through the respective *set requirement for aircraft* block to the *availability* block, requesting an aircraft be assigned for that mission. If an aircraft is available for tasking, the *availability* block will release it to meet

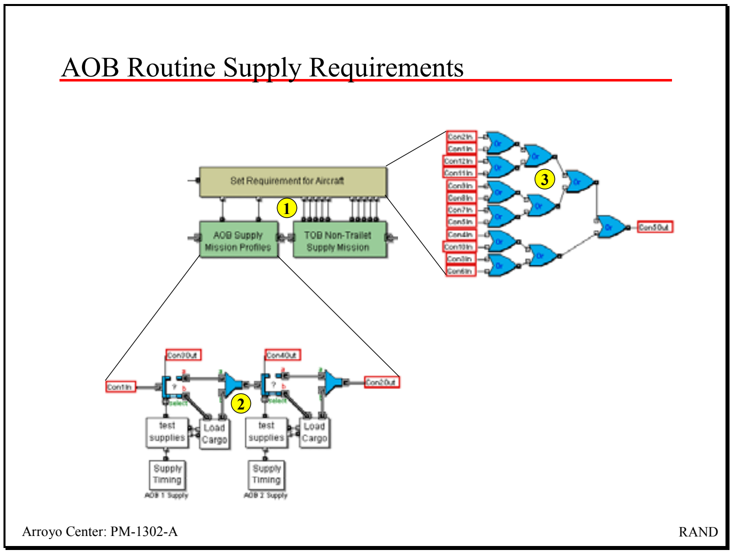
the request. Once the *availability* block has released an aircraft into the system it moves to the respective mission profile block where the aircraft is provided with an initial parameter set. All other parameters required to undertake the mission are set separately in the FOB parameter section, which is discussed later.



This slide describes the procedure for scheduling non-routine contingency missions. There are two parts to the process. The first is the non-routine mission profiles. Here mission tables are used to control the scheduling of the missions (see (1) and (2)). Each table represents a different mission type (*AOB casualty evacuation*, *TOB casualty evacuation*, *AOB emergency resupply*, *TOB emergency resupply*, and *TOB planning*). The structure of these tables is identical to those in the SFOB *set mission parameters* block.

The second part of the process is the *set requirement for aircraft* block. As is seen at (3), this block is a series of linked logical AND and OR nodes. The block receives inputs from the mission profile block (requests for aircraft) and outputs a ‘1’ if an aircraft is requested and a ‘0’ if not, to the *availability* block. If more than one aircraft is requested the output will remain a ‘1’ until all current requests are filled.

The mechanism for setting up a request for an aircraft through the *input* and *route test* blocks is identical to the mechanism performed in the SFOB and therefore is not described here.

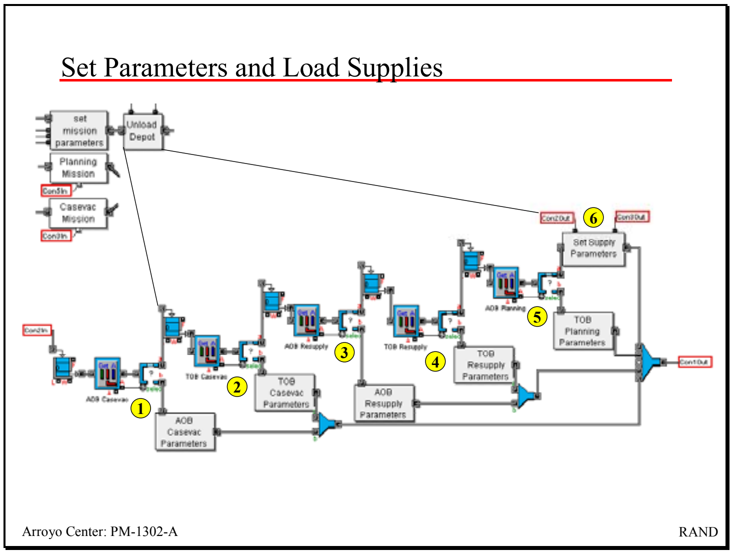


Like the procedure for queuing non-routine aircraft, the procedure for routine aircraft is a two-step process. The first is the *routine non-trailet supply mission* block at (2), which controls the release of supplies into the system, and therefore requests aircraft to conduct the supply missions. The second is the *set requirement for aircraft* block at (3), which is made up of connected logical OR gates that are used to send aircraft requests to the *availability* block. This second step is a simpler version of the *set requirement for aircraft* block that is used for non-routine missions.

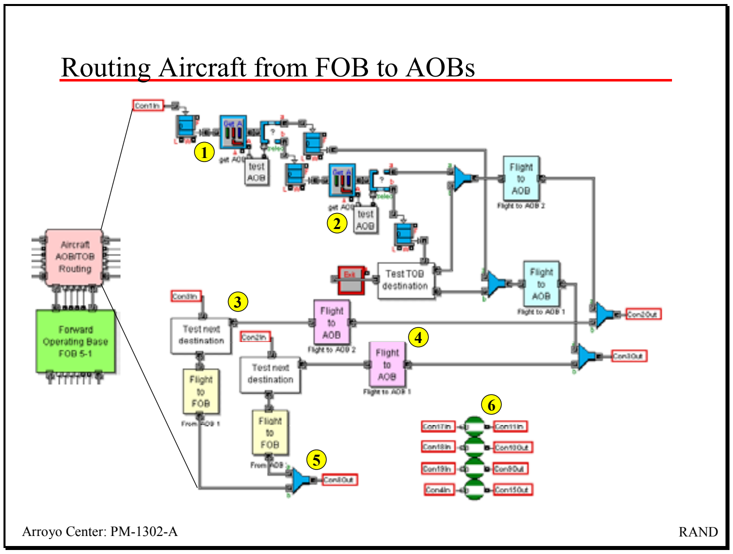
When supplies are release from the *supply-timing block* they are moved into the *test supplies* block, which registers that supplies are waiting for transport and requests an aircraft through the *set requirement for aircraft* to the *availability* block. Once an aircraft arrives, the *load cargo* block assigns the maximum amount of supplies possible to the aircraft. This amount however cannot exceed either the total amount available for transport or the total cargo capacity of the assigned aircraft. Since an aircraft may hold more supplies than the requirement for a single base, if supplies for other bases have been released, the aircraft will visit the *load cargo* blocks for those bases and load supplies up to its remaining capacity. Once an aircraft is fully loaded or no additional supplies are required, the aircraft is

released from the mission area. If there are additional supplies still requiring transport, a second aircraft will be released from the *availability* block.

A similar approach to the above is used to transport supplies from the FOB to the SOF teams (TOB).



The *set parameters and load supplies* block performs similar functions to the *set parameter* blocks in the SFOB. The AOB and TOB casualty evacuation parameters are set at (1) and (2) respectively. The AOB and TOB emergency resupply parameters are set at (3) and (4), and the TOB planning parameters are set at (5). Finally, the routine supply parameters are set at (6).

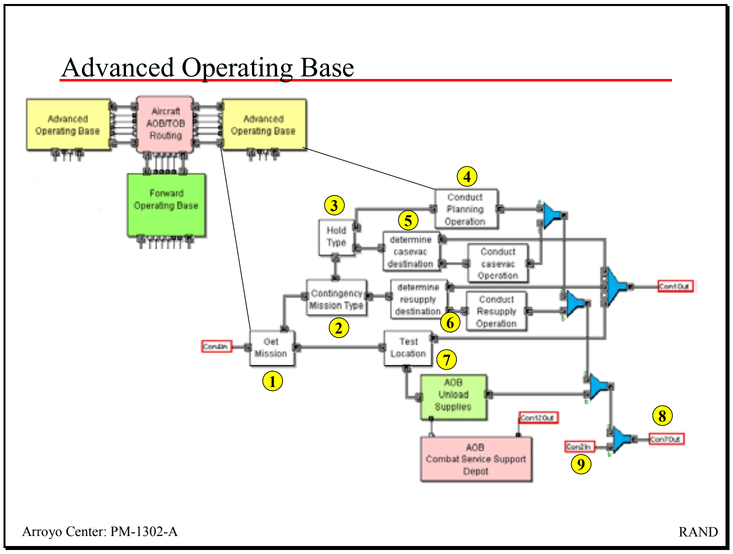


This slide describes the mechanism by which aircraft are routed from the FOB to the advanced operating bases and SOF teams and then back.

Aircraft enter the block from the FOB through the *Con1In* control at (1) and (2). Upon entering, a test is performed to determine the first AOB assigned to the aircraft. When a destination is determined, the aircraft is sent to the corresponding *flight to AOB* block (shown in blue). The contents of these blocks are identical to those in the *aircraft SFOB-FOB routing* and so are not describe any further here.

Aircraft returning from an AOB enter the block through one of the inputs (depending on the AOB) at (3). Here the aircraft is tested in the *test next destination* block to determine whether it has completed its entire set of assigned missions or if it is required to visit other bases. In the case where it is required to visit another advanced base, the aircraft is flown to that base through one of the pink *flight to AOB* blocks at (4). If no additional advanced bases require visits by the aircraft, it is sent back to the FOB through one of the *flight to FOB* blocks (shown in yellow) at (5). The information links at (6) route information between the FOB and each AOB.

Advanced Operating Base (AOB)



There are two types of advanced operating bases, the one shown in this slide (which is the simpler) and one that is similar in structure to an FOB (i.e., one in which aircraft are allocated to the base and the base is tasked with maintaining the operational status of its lower echelon bases). Due to the similarity of the second type of AOB to the FOB, it is not discussed here.

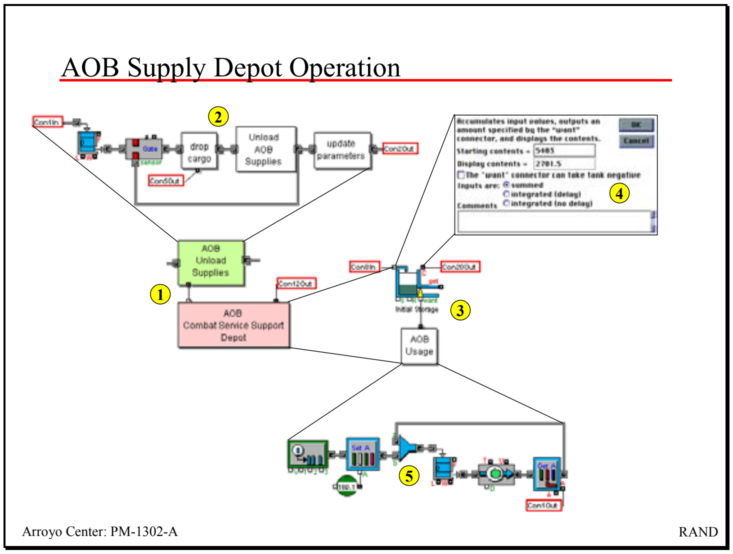
Aircraft arrive at the AOB from the FOB through the *Con4In* connector. Upon arrival, the *get mission* block at (1) determines whether the mission type is routine or non-routine. Non-routine missions are routed to the *contingency mission type* block at (2) where the mission type is further broken down into the individual contingency types.

Casualty evacuation and planning mission types are routed through the *hold type* block at (3) to their respective mission blocks at (4) and (5). However, since casualty evacuations can be conducted to AOBs and to SOF team locations, the *determine casevac destination* block at (5) is used to distinguish between the destinations. If the casualty evacuation mission type is to one of the SOF teams, the aircraft is directed to that team through the *Con1Out* connector. The *conduct casevac operation* block is used to conduct casualty evacuations at the

FOB. Emergency resupply contingency missions are processed in a way that is similar to that of the casualty evacuation missions at (6).

If the mission is determined at (1) to be a routine supply mission, the aircraft is routed to the *AOB unload supplies* block at (1). Here the supplies for this particular base are unloaded into the base combat service support depot and the aircraft is returned to the *FOB-AOB aircraft routing* block through the output at (8). The unload supplies procedure and the contents of the depot is described on the next slide.

Aircraft returning to the AOB from one of the SOF teams enter the block at (9) and are immediately routed to the *FOB-AOB aircraft routing* block.



When an aircraft enters an AOB with a routine supply mission, it is routed to the *test location* block shown at (7) on the previous slide. This block is used to determine whether the supplies are for the AOB or for one of the lower level SOF teams. If the supplies are for an SOF team, the aircraft is routed out of the AOB to the particular SOF team. If the supplies are for the AOB, the aircraft moves to the *AOB unload supplies* block shown at (1) and in detail at (2).

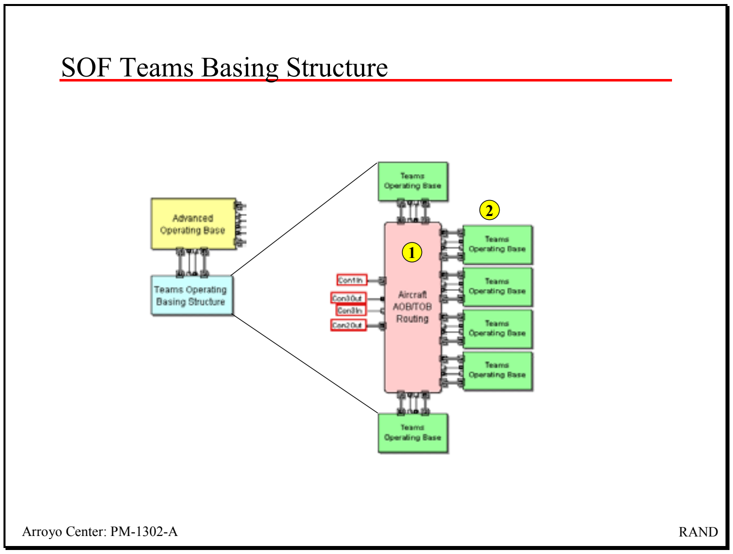
The *AOB unload supplies* block processes aircraft on a first-come first-served basis. The *gate* block ensures that only one aircraft is processed at a time. At (2) the *Con5Out* connector is used to move the supplies from the aircraft to the AOB supply depot shown at (3).

The depot consists of a supply storage tank. Supplies enter through the *Con8In* connector and are consumed in the *AOB usage* block shown at (5). The insert at (4) illustrates the initial supplies and current supplies in storage. The *Con20Out* connector is used to transfer information about the status of the storage depot to the reporting section.

The *AOB usage* block at (5) periodically (usually 8 hours) depletes the depot supplies by a fixed amount representing the supplies that would ordinarily be consumed by the base during that period. The block operates by creating a single simulation item at the start of the

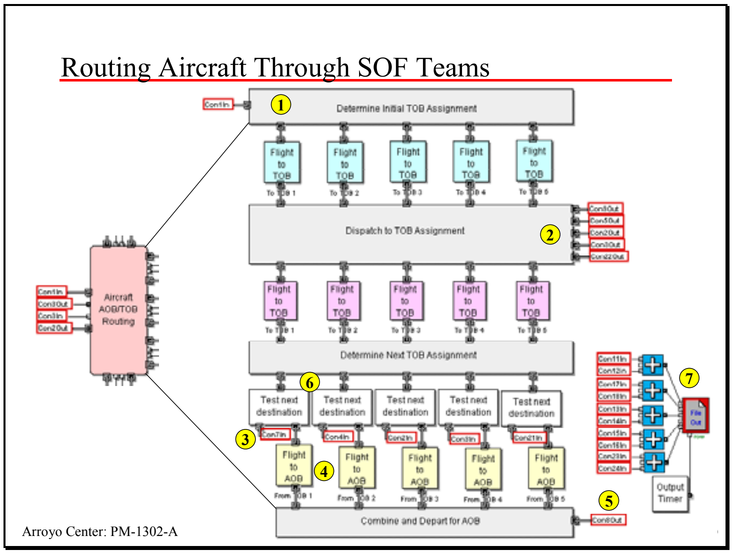
simulation. This item is assigned a predetermined consumption value in the *set attribute* block. A *constant* block is used to provide this value, (180.1 in the example shown). The item then moves into a continuous 8-hour loop. During each loop the *get attribute* block is used to remove supplies from the depot.

Special Operating Forces Teams
(TOB)



At the lowest level of the structure are the special operating force (SOF) teams. These teams, usually prearranged in groups of five, can either be administered by the AOB directly above them in the command chain, or by the FOB, which is two levels above them. Either way, the teams operating basing (TOB) structure shown in the slide is unaffected.

The TOB structure consists of an aircraft routing block at (1) and up to five SOF teams at (2). The *aircraft AOB-TOB routing* block receives aircraft and coordinates their movement between the SOF teams. The SOF teams fulfill the mission (contingency or supply) assigned to the arriving aircraft, and consume supplies.



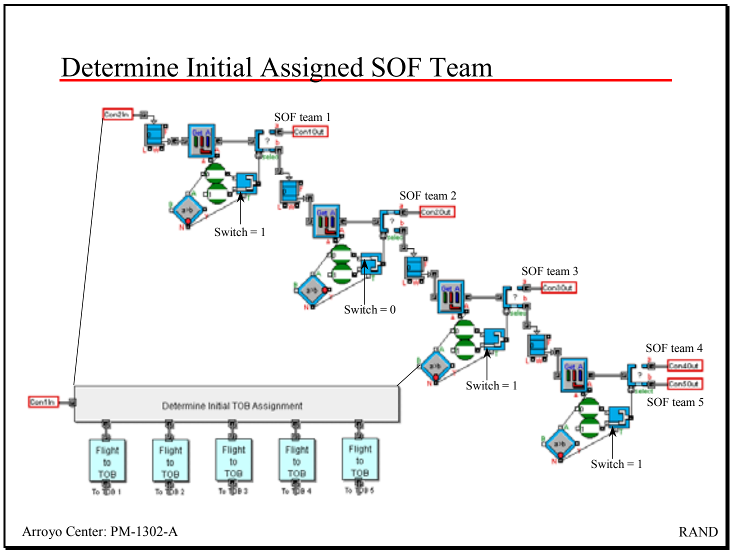
The *aircraft AOB-TOB routine* block is essentially no different from either of the routing blocks discussed thus far. As with the others, the difference is the number of bases the routing block controls. In this case it is the five SOF teams.

Aircraft arriving from higher echelon bases arrive through the *Con1In* connector at (1). Here aircraft are tested in sequential order to determine their first assigned SOF team (see the next slide for details). Once a team is determined, the aircraft moves into the appropriate *flight to TOB* block (shown in blue). The aircraft then exits the routing block through one of the connectors at (2) and heads toward its specified SOF team.

Aircraft arriving at the routing block after visiting an SOF team enter through one of the connectors at (3). The actual connector used depends on which SOF team was visited. The arriving aircraft here is then tested in the *test next destination* block at (6) to determine whether or not it has completed its entire mission. If it has, it is sent back to the AOB through the corresponding *flight AOB* block and the *Con8Out* connector at (5). If the aircraft has not completed its entire mission, it is moved to the *determine next TOB assignment* block where the next SOF team to visit is found. Once found, the aircraft moves to the appropriate *flight to TOB* block (shown in pink), which simulates the

flight to the next SOF team. The aircraft then leaves the routing block through the appropriate connector at (2).

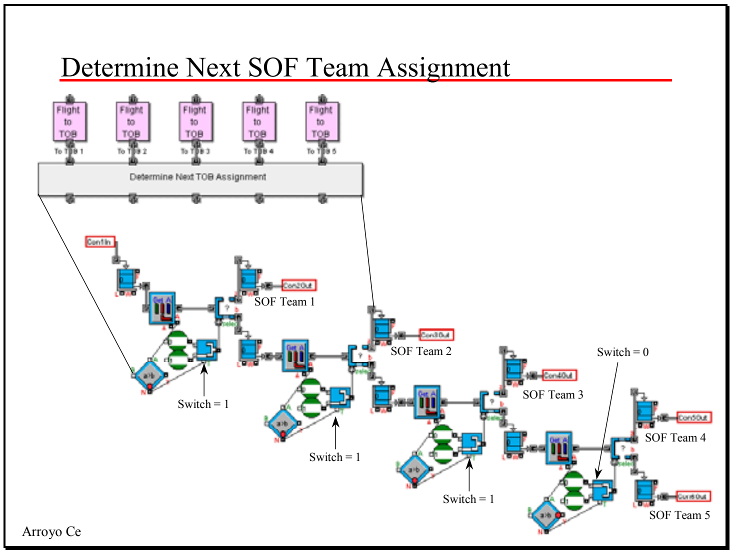
Data relating to the content of the storage depots at each of the SOF teams is monitored at (7). The information collected here can be sent directly to an output file for further analysis and/or displayed on the screen during the execution of the simulation.



When aircraft arrive at the *aircraft AOB-TOB routing* block a determination is made as to their first assigned SOF team.

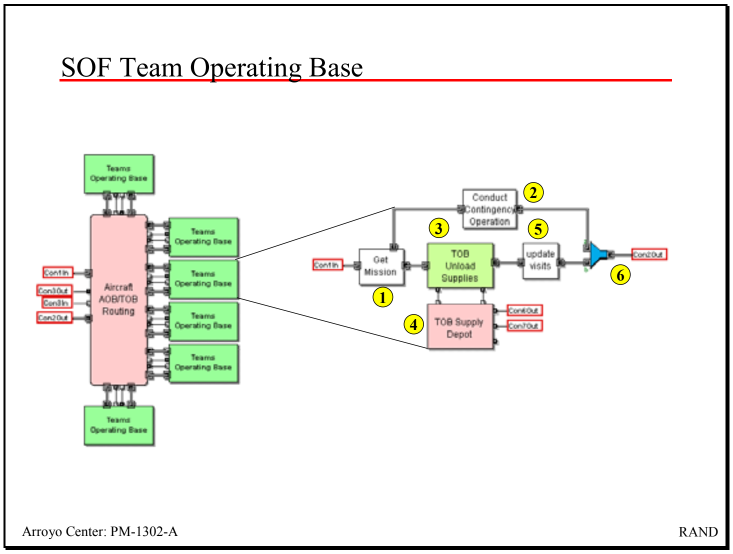
Here the arriving aircraft's parameters associated with the SOF teams are tested sequentially until a SOF team is found. Once a team is found the decision switch associated with that particular team is changed from '1' to '0' effectively routing the aircraft to that team. If the test for SOF team 4 fails, the procedure assumes that the assigned SOF team is team 5.

The illustration in the slide indicates that the current aircraft will be routed to SOF team 2.



The structure to determine the next SOF team to be visited by the aircraft is similar to what is used to determine the first SOF team destination on the previous slide.

Here, aircraft enter through the *Con1In* connector and are tested sequentially until the next SOF team is found. The example shown in the slide indicates that the next SOF team to be visited is Team 4, since the switch relating to that team is set to ‘0’.



The SOF team-operating base is the lowest level of the model. Aircraft that arrive at this base through the *Con1In* connector are tested in the *get mission* block at (1) to determine whether their mission is contingency or supply. Contingency missions are routed to the *conduct contingency operation* block at (2) and supply missions are routed to the *unload supplies* block at (3).

The *conduct contingency operation* block consists of the simple *wait* block that hold the aircraft for the preset time and a *set attribute* block that resets the contingency mission parameters indicating its successful completion. The *unload supplies* block at (3) transfers the supplies from the aircraft to the supply depot at (4). Once completed, the *update visit* block at (5) resets the supply parameters to indicate the mission has been completed.

Once the mission is completed the aircraft is returned to the *AOB-TOB routing* block through the *Con2Out* connector at (6).

SECTION THREE

Preparing a SOFAADM Simulation

Arroyo Center: PM-1302-A RAND

In preparing a SOFAADM simulation, specific data representing the Special Force Group implementation must be entered into several of the modules. This section describes the process analysts should adhere to when developing and running a SOFAADM simulation.

Preparing a SOFAADM Simulation

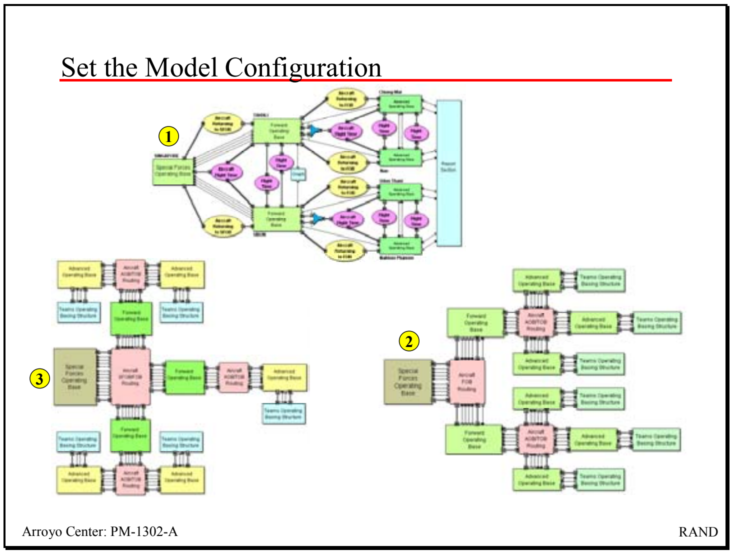
1. Set the model configuration
2. Set aircraft type parameters
3. Set the non-routine contingency mission profiles
4. Set the routine supply parameters, including delay times
5. Set the distance parameters
6. Set FOB, AOB, and SOF Team supply consumption rates
7. Set the simulation output requirements
8. Run the Simulation

Arroyo Center: PM-1302-A RAND

There are essentially eight steps that must be undertaken before a simulation should take place.

1. Set the model configuration
2. Set aircraft type parameters
3. Set non-routine contingency mission profiles
4. Set routine supply parameters, including delay times
5. Set distance parameters between bases
6. Set FOB, AOB and SOF Team supply consumption rates
7. Set simulation output requirements
8. Run the simulation

The following slides describe each of these steps.

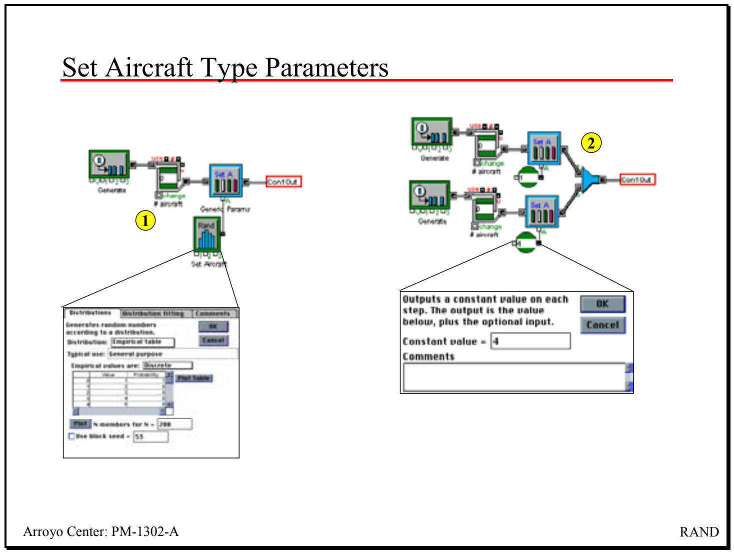


Here the variant of the model the analyst is using must be configured to represent the structure of the Special Force Group (SFG) being simulated. This step is by far the longest and the most difficult. The problem is that most of the SFGs operate differently with different command structures and different supply processes. The model must be configured to accommodate all of these differences.

This slides illustrates the differences between three SFGs. The structure at (1) is the simplest of the three. The configuration here includes two forward bases, each of which administers two advanced bases. There are no SOF teams explicitly modeled in this variant. The variant at (2) again has two forward bases, but each of these now administers three advanced bases, each of which in turn administers five SOF teams grouped into a teams operating base module. The variant at (3) contains three forward bases each of which administers up to two advanced bases and a number of SOF teams.

If the SFG to be modeled is similar in basic structure (same number of forward, advanced and SOF team bases and similar supply requirements) to the variant currently used, the analyst should only have to modify the model's internal parameter sets to represent the new SFG (see steps 2-6). If however, major changes are required, it is recommended that

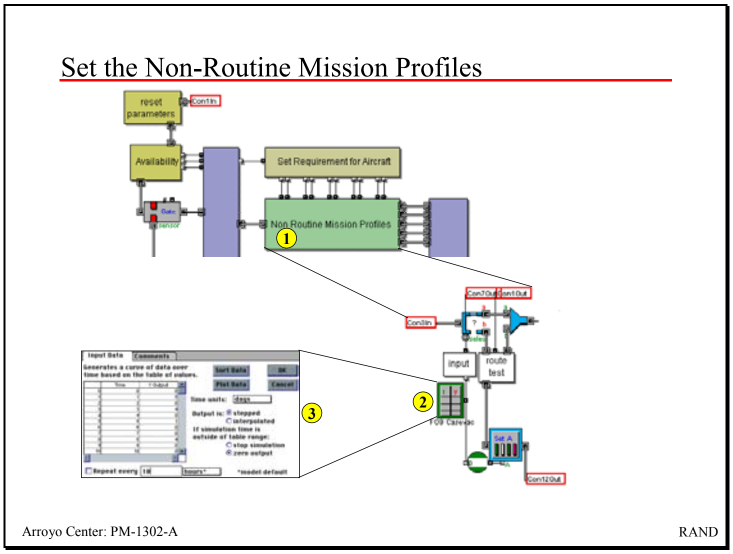
the analyst build a new model variant using the component modules of the current variant as building blocks.



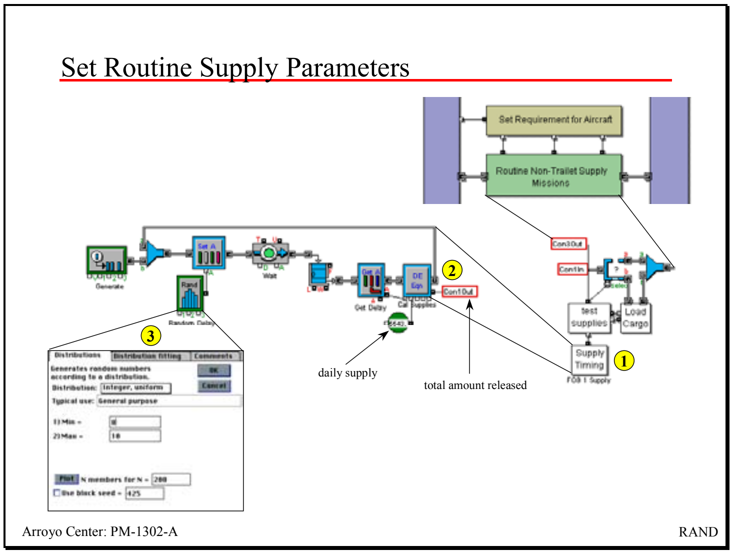
There are two ways in which the aircraft type can be set.

1. If a single aircraft type is to be used or analysts wish to randomize the selection between the five default types, then the configuration at (1) is recommended. Here the *set attribute* block connector is linked to an *input random number* block, which is used to determine the aircraft type.

2. If multiple aircraft types are required and analysts want to force the determination of specific types, then the configuration at (2) should be implemented. In this case, a series of configurations similar to those at (1) are linked in parallel and a *constant* block is used to determine the aircraft type instead of a random block. The example shown in this slide illustrates that two aircraft type will be created, type 1 Chinook CH-47 and type 4 Sherpa C-23.



This slide describes the procedure for implementing non-routine contingency mission profiles. Here the mission tables that are used to control the scheduling of the missions (see (1) and (2)) must be configured to represent the frequency of the specific mission type. Although only one mission type is shown in the slide, there are three basic mission types (*casualty evacuation*, *emergency resupply*, and *planning*). The structure of a typical table is shown in the insert at (3). Within each table the *time units* is given in days. Consequently, the '*Y Output*' values vary after every 24-hour period of simulation real-time. The values themselves represent the base in which the mission is to be conducted. A zero entry is used to represent the case where no mission is being scheduled. Finally, analysts must place a value for each day of the simulation period.

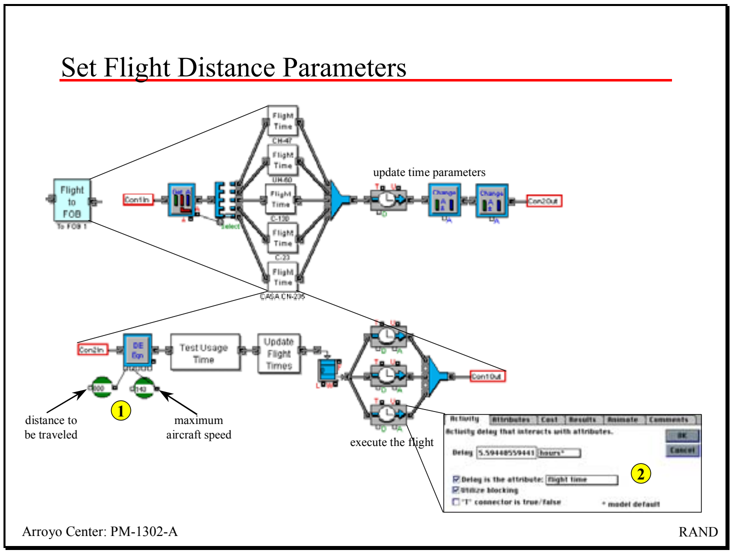


The *supply-timing* block at (2) is used to control the release of supplies into the system. This block functions in isolation to the remainder of the simulation. Its only connection to the simulation is the *Con1Out* output connecter, which sends a value representing the amount of supplies to be released to a particular base to the *test supplies* block.

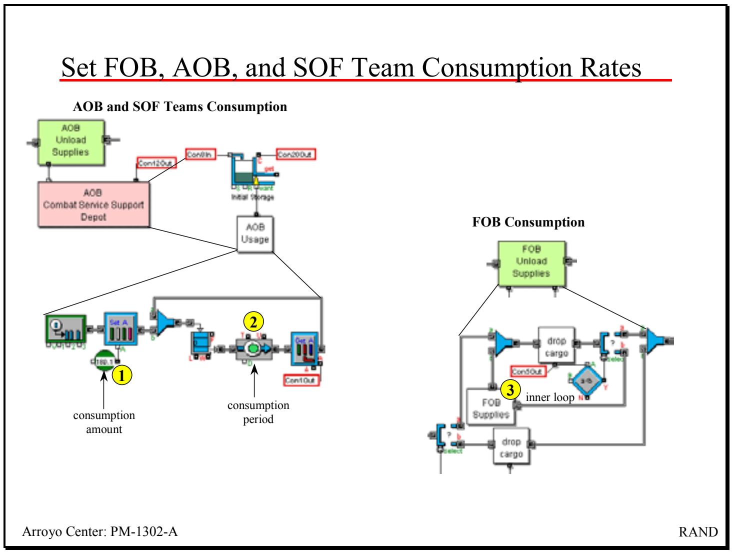
Within the *supply-timing* block, the *set attribute* block and the *random delay* block combine to set the timing for the release of supplies. The random parameter at (3) represents the number of days that must pass before supplies can be released. The insert illustrates that the delay value is currently an integer value between 8 and 10 inclusive. Consequently, between 8 and 10 days must pass before the next amount of supplies is released. Analysts vary the release of supplies by adjusting the *min* and *max* integer values.

Since the aim of the model is to sustain the bases, the amount of supplies released must equal the amount consumed during the wait period. To calculate the required supplies, the daily-consumed supply is multiplied by the delay time. Analysts need to enter the daily supply requirement into the *constant* block connected to the *equation* block.

These adjustments must be copied to all the *supply-timing* blocks.



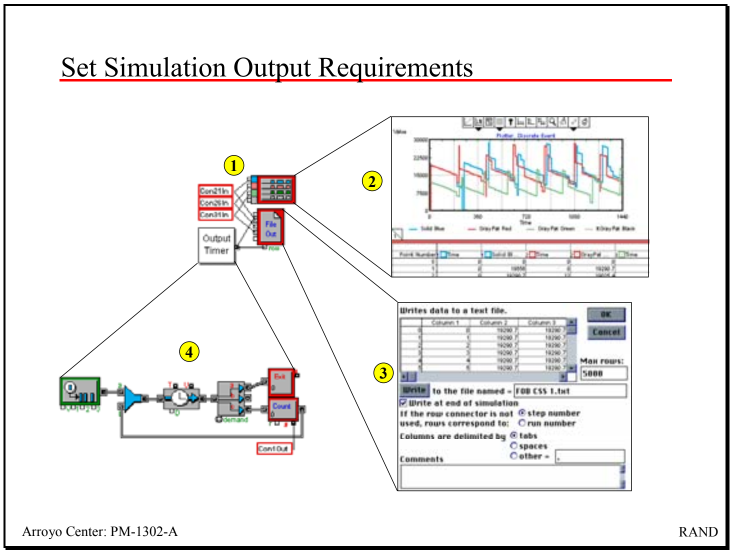
This slide describes a typical *flight* block. All flight blocks are similar to the one shown. The only difference between these blocks is the distance that needs to be traveled. This parameter shown at (1) must be updated to reflect the current basing structure. The example in this slide indicates that the distance between two bases is 800 nm. Since this value is used to calculate the flight time (at (2)) between two specific bases, it must be updated in all instances of the *flight* block.



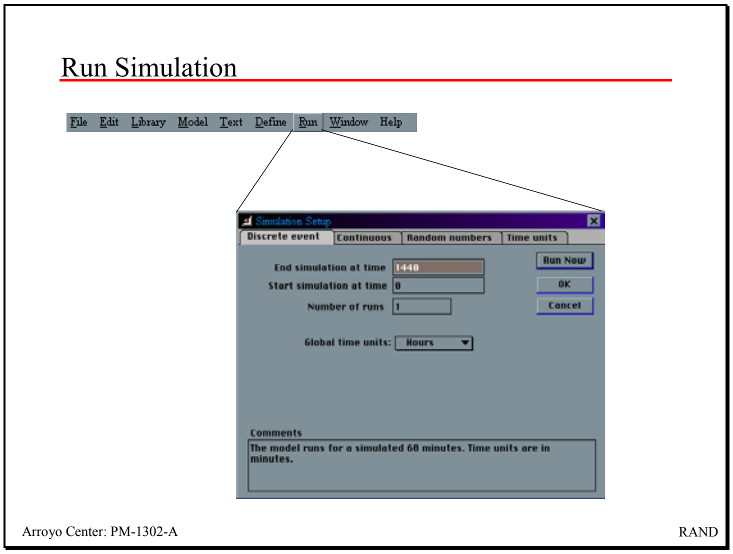
There are two types of base consumption parameter sets that need to be updated. The first is for consumption at the AOB and SOF teams, and the second is for consumption at the FOB.

For the AOB and SOF teams, consumption parameters are found within the *usage* block, which is part of the *combat service support depot* block. The consumption amount is entered into the *constant* block at (1) and the consumption period is entered into the *wait* block at (2). In the example shown 180.1 lbs will be consumed by the base during each period. The consumption period is usually set at 8-hours.

For the FOB, the consumption parameter is found within the *FOB unload supplies* block at (3). This block is similar to that of the consumption block for the AOB and SOF teams, however since this block interacts with the supply depot through the *drop cargo* block a negative value must be used. If a positive value is used the result would be an increase in total supplies rather than a decrease.



The setup of a typical output is shown at (1). The amounts of supplies in the depots enter through the control inputs. These values are sent to both the simulation graphical display shown at (2) and the *file out* block at (3). An *output timer* block shown at (4) is used to synchronize the output to file. This block is a continuous loop that outputs an hourly count to the output file.



The final step is to run the simulation. The simulation setup dialog is displayed in this slide. Analysts need to enter values for the four discrete event parameters, *end simulation at time*, *start simulation at time*, *number of run*, and the *global time units*. Once values for each of these have been entered the *run now* button is used to commence the simulation.

SECTION FOUR

Case Studies

Arroyo Center: PM-1302-A RAND

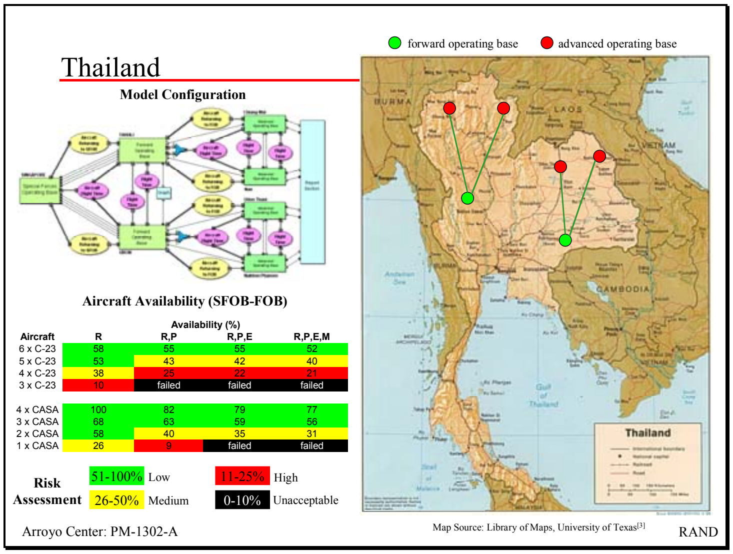
In this section three case studies are presented: Thailand, Colombia, and the Congo. These studies are for illustrative purposes only; they are not intended to represent Special Forces deployments in any way.

For each study the SFG model configuration is presented along with a map of the area indicating notional locations of forward and advanced bases, and an example set of results for SFOB-FOB aircraft.

For each of the studies the following experimental procedure was used.

1. Establish a baseline fleet size for each of the aircraft types. A baseline simulation is one in which only routine supply missions are conducted. Baseline results are presented in column ‘R’ in the result section.

2. Once a baseline fleet is established the non-routine planning, emergency resupply, and casualty/medical evacuation missions (represented as ‘P’, ‘E’, and ‘M’ or ‘C’ in the result section) are added sequentially. The measure of effectiveness is the aircraft availability variable.



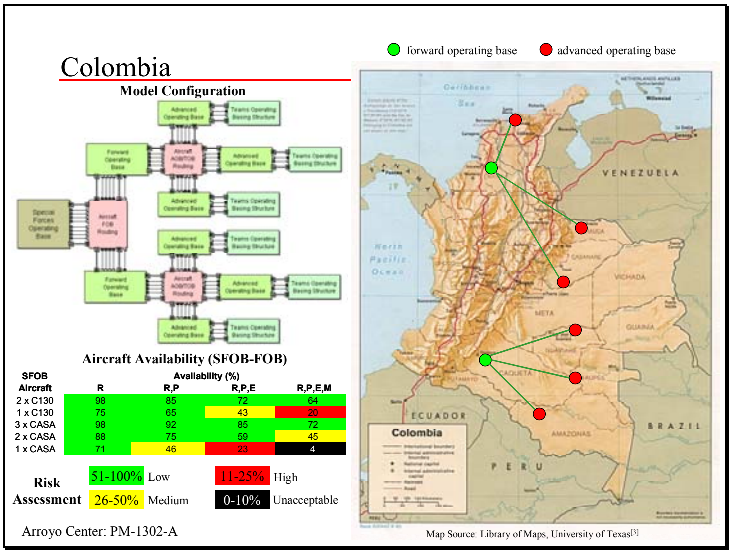
This slide illustrates a hypothetical Special Forces deployment to Thailand. One of the first considerations is to configure the model so that it captures the organizational and geographical nature of the intended operation. In this case, we have a map depicting where desired FOBs would be located (Takhli and Ubon) and the AOBs that will be supported (Chiang Mai, Nan, Udon Thani, and Nakhon Phanom). The model configuration shown above captures this structure of an SFOB at Singapore supporting two FOBs and then each FOB supporting 2 AOBs. From this information the modeler also needs to know the distance between the nodes so that flying time can be calculated internally to the model.

Additional characteristics of the Special Forces deployment that the modeler needs to know are the number of soldiers by location, the frequency and details of non-routine missions, and the operational data of feasible aircraft alternatives. Partial results of running this particular model are shown above in the Aircraft Availability section. The results are for the SFOB to FOB linkages with the key measurement being the percent of time at least one aircraft is available for tasking. From the chart we see the fleet sizes of two different aircraft, i.e., C-23 Sherpas and CASAs. The first column, “R”, is when the tasking is only for routine supply missions. The level of availability required is a subjective risk assessment that a commander

must establish. If a commander decided that availability must be at least 50%, then the minimum fleet size is 5 C-23s or 2 CASAs. If the commander was comfortable down to a 35% availability level, a fleet size of 4 C-23s would be sufficient. Again this is a subjective assessment based on individual parameters. Since all the availability numbers in the column are positive, the basic missions can be accomplished with even the smallest fleet size, i.e., 3 C-23s or 1 CASA. However, it is likely that a commander will want some reserve capability to meet unforeseen requirements.

Next non-routine or contingency missions are considered by adding them one at a time to the mission taskings. Thus the second column, “R,P”, is where non-routine planning missions have been added to the usual routine resupply missions. If the commander’s risk level remained where he required at least 50% availability, then the basic fleet size would have to increase by one C-23 to a total of 6 (or by one CASA to total of 3). When the block is labeled “failed” this means the missions could not be met with that fleet size.

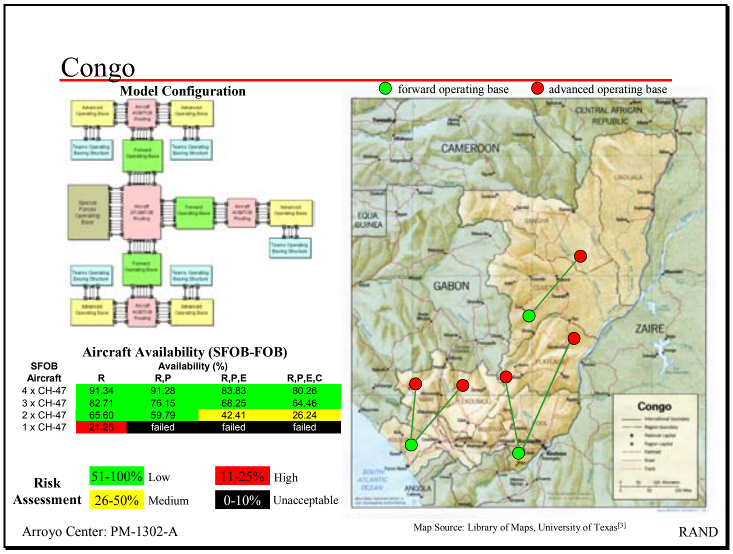
Moving across the columns to the right we incrementally add contingency missions of emergency resupply, “E”, and casualty/medical evacuation, “M”, to the mission mix. Given the possibility of all these contingency missions being possible along with a risk level of at least one aircraft being available 50% of the time, the required fleet size would be 6 C-23s or 3 CASAs.



This slide illustrates a hypothetical Special Forces deployment to Colombia. The chosen base structure depicted here includes an SFOB linked to two FOBs, which in turn administer three AOBs each. As in any particular regional scenario, the geographical layout of bases will be driven by terrain and available airbases. We see here that the distances between the bases varies quite a bit.

Additional characteristics of the Special Forces deployment that the modeler needs to know are the number of soldiers by location, the frequency and details of non-routine missions, and the operational data of feasible aircraft alternatives.

Partial results of running this particular model are shown above in the Aircraft Availability section. The results are for the SFOB to FOB linkages with the key measurement being the percent of time at least one aircraft is available for tasking. From the chart we see two different aircraft are modeled--C-130s and CASAs. In the first column, we see that either one C-130 or one CASA is sufficient down to 70% availability when only the routine supply missions are considered. Factoring in all three types of contingency missions, two C-130s or 3 CASAs would be required to maintain at least 50% availability.



SFOB Aircraft	Availability (%) R	R,P	R,P,E	R,P,E,C
4 x CH-47	91.34	91.28	83.83	80.26
3 x CH-47	82.71	76.15	68.25	64.46
2 x CH-47	65.60	59.79	42.41	26.24
1 x CH-47	21.25	failed	failed	failed

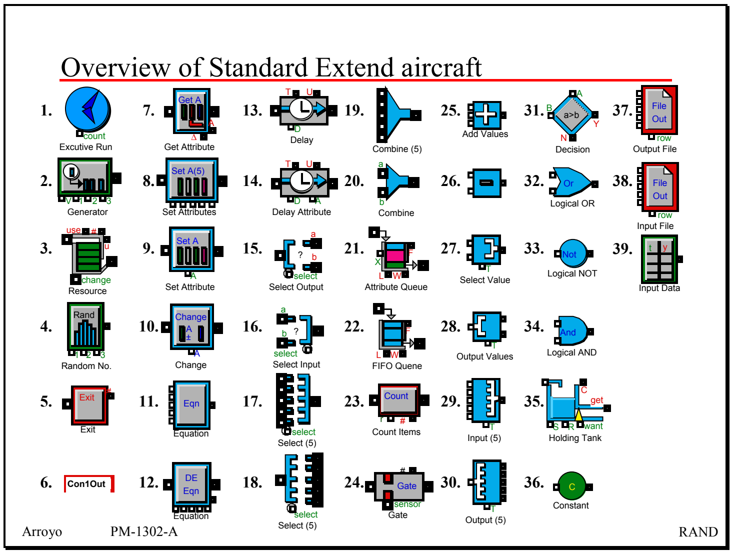
The third case study depicts a hypothetical Special Forces deployment to the Congo. The chosen base structure depicted here links an SFOB to three FOBs. Two of the FOBs administer two AOBs each, while the third FOB supports a single AOB. As in any particular regional scenario, the geographical layout of bases will be driven by terrain and available airbases. The challenge here is the SFOB supporting three dispersed FOBs.

Additional characteristics of the Special Forces deployment that the modeler needs to know are the number of soldiers by location, the frequency and details of non-routine missions, and the operational data of feasible aircraft alternatives. In this scenario only the C-47 is considered.

Partial results of running this particular model are shown above in the Aircraft Availability section. The results are for the SFOB to FOB linkages with the key measurement being the percent of time at least one aircraft is available for tasking. In the first column, we see that two C-47s are needed to stay above 50% availability when only the routine supply missions are considered. To meet all three types of contingency missions, an additional C-47 would be required to maintain at least 50% availability.

SECTION FIVE

Appendices



Taken from *Extend™ Simulation software for the next millennium* (see references), the following are descriptions of some of the most commonly used Extend™ modeling blocks in the SOFAADM.

1. The *executive block* is the heart of the discrete event model and must be placed to the left of all other blocks. The *executive block* allows the duration of the simulation to be controlled by the end time or by another number that can specified in the executive block dialog.
2. The *generator block* provides items for the simulation at specified inter-arrival times. In the simulation SOFAADM all items are created at the start of the simulation using this block and the *resource block* described next.
3. The *resource block* holds and provides items to be used in the simulation. It is used as part of the open system. The block is used to set the initial number of aircraft in the model.
4. The *input random number* generates integers or real numbers based on the selected distribution. In the SOFAADM this block is used to specify the aircraft type entering the system.
5. Items to be passed out of the simulation are sent to an *exit block*. The total number of items absorbed by this block is reported in its dialog.

6. This *control block* allows the passing of items between simulation modules.
7. The *get attribute block* displays and/or removes attributes in items, then passes the item through. The attribute value is in the dialog and output to the A connector.
8. The *set attribute (5) block* is used to set the attributes of items passing through it. Up to five attribute names and values may be assigned to an item with this block.
9. The *set attribute block* sets the attributes of items passing through it. Up to seven attribute names and values may be assigned to an item with this block. However, unlike the set attribute (5) block, only one attribute can be set using the A connector.
10. The *change attribute block* is used to change an item's attribute value then pass the item through. The attribute to be changed is specified by the user. Attributes values can either be incremented, decremented, multiplied or divided.
11. The *equation block* outputs the result of an equation entered in the dialog. The equation must be of the "Result = formula".
12. The *DE equation block* calculates an equation when an item passes through it. The inputs to the equation can be from the item's attributes, values, priority, or from one of the five value input connectors. The result of the equation can be optionally assigned to an attribute.
13. The *activity delay block* holds an items for a specified amount of time, then releases it. The delay time is the value in dialog, or if connected, the value at the D connector when the item is received. The connector overrides the dialog.
14. The *activity delay attribute block* works the same as the activity delay block described in (13) except it interacts with an item's attributes.
15. The *select DE output* block selects the input item to be output at one of two output connectors based on a decision. The item at the input is passed through the selected output.

16. The *select DE input block* selects one input to be output based on a decision. The item that is present at the selected input is passed through the output.
17. The *select DE input (5) block* is similar to (15) except it allows the output to be selected from up to five inputs.
18. The *select DE output (5) block* is similar to (16) except it allows the input to be passed to up to five outputs.
19. The *combine (5) block* combines the items from five different sources into a single stream.
20. The *combine block* combines the items from two different sources into a single stream.
21. The *attribute queue block* is a queue where items with a particular attribute have a higher priority than other items. If there are no attributes to prioritize, this becomes a simple first-in-first-out (FIFO) queue.
22. The *FIFO queue block* is a queue where items are passed through on a first-in-first-out basis.
23. The *count items block* passes items through and reports the total number of items passed in its dialog and at the # connector.
24. The *gate block* allows a specified number of items to be in a section of the model at any one time. This block is used to restrict the passing of items into a system that allows only a specified number of items in that system at any time.
25. The *add block* sums the values at the three inputs and outputs the total.
26. The *subtract block* subtracts the bottom input from the top input and outputs the result.
27. The *select input block* selects its output to be either of the two inputs based on a threshold test. The block essentially acts like a switch. The value to be output is determined by comparing the value of the T connector to a critical value in the dialog. When this value is less than the critical values, the top input is used, otherwise the bottom input is used.

28. The *select output block* passes the input value to one of two output connectors based on a threshold test. The output connector is selected by comparing the value of the T connector to critical value that is specified in the dialog. When the value of the T connector is less than the critical value, the top output is selected, otherwise, the bottom output is selected.
29. The *select input (5) block* selects its output to be one of five inputs according to the value of the T connector. The top input is selected if the T connector is 1, and the bottom input is selected is the T connector is 5.
30. The *select output (5) block* passes the input value to one of five outputs according to the value of the T connector. The top output is selected if the T connector is 1 and the bottom output is selected if the value is 5.
31. The *decision block* makes decisions based on the inputs and internal logic defined by the user. The dialog allows the following test comparing A to B: greater than, greater than or equal to, equal to, less than, less than or equal to, and not equal.
32. The *logical OR block* performs the logical OR operation on its inputs. If either of the inputs is greater than 0.5, the output is 1, otherwise, the output is 0.
33. The *logical NOT block* performs the logical NOT operation. If the input is greater than 0.5, the output is 0, otherwise, the output is 1.
34. The *logical AND block* performs the logical AND operations. If each of the two inputs is greater than 0.5, the output is 1, otherwise, the output is 0.
35. The *holding tank block* accumulates the total of the input values, and allows a requested amount to be removed, if it is available.
36. The *constant value block* generates a constant value at each step. Users specify the constant value in the dialog (the

default constant is 1). If the input is connected, the input value is added to the constant in the dialog.

37. The *file output block* writes data from the block to a text file. Data can be pasted into the block, or model data can be input through the input connectors during the simulation. The file name can be either entered before the simulation or left blank. If the fine name is left blank the model will prompt for a file name at the conclusion of the simulation.
38. The *file input block* reads data from a text file and write it into the block’s table. Once the data is in the table, it can be used in the model.
39. The *input data block* generates a curve of data over time from a table of values and acts as a lookup table.

The reader is reminded that the above are only brief descriptions of the model blocks. For a more comprehensive description please refer to the Extend documents listed in the References.

Commonly Cited Output Variables

1. Average aircraft queue length.
2. Average aircraft wait time for scheduling.
3. Maximum aircraft queue length.
4. Maximum aircraft wait time for scheduling.
5. Current number of aircraft waiting for scheduling.
6. Total number of aircraft arrivals.
7. Total number of aircraft departures.
8. Percentage of time there was a least one aircraft available.
9. Frequency of non-routine contingency missions.
10. Percentage increase in aircraft workload over routine supply missions.

Arroyo Center: PM-1302-A RAND

A number of output variables are commonly used; definitions are self-explanatory.

Commonly Used SOFAADM Parameters

1. a1 – a6
2. amedivac
3. aresupply
4. atype
5. daily hours
6. f1 – f3
7. flight hours
8. flight time
9. fmedivac
10. medivac
11. mindex
12. origin
13. pindex
14. planning
15. random delay
16. residual cargo
17. resupply
18. rindex
19. supply
20. t1 – t10
21. tmedivac
22. total time
23. tresupply
24. type
25. wait time

Arroyo Center: PM-1302-A RAND

1. a1 – a6: These are used as indices for the advanced operating bases. Depending on your implementation you may have to increase this number. These variables are used for two functions, (1) to indicate that an AOB must be visited as part of a contingency mission, and (2) to represent the amount of supplies that needs to be transported to that particular base.
2. amedivac: The amedivac variable takes values ‘0’ or ‘1’ and indicates that a casualty evacuation mission to an advanced operating base has been scheduled. When amedivac is set to ‘1’ the model will interrogate the variables a1 – a6 to determine which of the bases contains the casualty.
3. aresupply: The aresupply variable works the same as for amedivac. The only difference is that aresupply indicates that an emergency resupply mission has been scheduled instead of an casualty evacuation. Again a1 – a6 is used to determine which base requires the supplies.
4. atype: This variable indicates the type of aircraft that is being used. The default model has five aircraft types available, Chinook CH-47, Black Hawk UH-60, Hercules C-130, Sherpa C-23 and the CASA CN-235. The simulation represents these by the indices 1 – 5 respectively.

5. daily hours: This variable indicates the maximum number of flight hours per 24 hour period an aircraft undertake. Once this value has been exceeded the aircraft must stand-down for the remainder of the period.
6. f1 – f3: These variables are used as indices for the forward operating bases. Depending on your implementation you may have to increase this number. These variables are used for two functions, (1) to indicate that an FOB must be visited as part of a contingency mission, and (2) to represent the amount of supplies that needs to be transported to that particular base.
7. flight hours: This variable is a cumulative variable for the number of flight hours an aircraft has undertaken in a 24-hour period. At the commencement of each 24-hour period this value is reset to zero. This variable is used in conjunction with *daily hours* to determine if an aircraft has exceeded its flying limit.
8. flight time: The flight time variable holds the actual amount of time an aircraft must fly between bases.
9. fmedivac: This variable is a binary variable. By default it is set to ‘0’. It is set to ‘1’ if a casualty evacuation mission is queued to an FOB.
10. medivac: This variable is a binary variable and its default value is ‘0’. The variable is set to ‘1’ if a casualty evacuation mission is queued.
11. mindex: This variable indicates either from which FOB or which AOB the casualty evacuation is to occur.
12. origin: This variable indicates the originating base of an aircraft. It can take one of three values, ‘0’ indicates the aircraft originated at the SFOB, a ‘1’ indicates an FOB aircraft, and a ‘2’ indicates an AOB aircraft.
13. pindex: The variable indicates the location of the planning meeting. The variable usually indicates an FOB or an AOB.
14. planning: The planning variable is binary. A ‘0’ indicates that no planning mission is scheduled and a ‘1’ indicates that a planning mission is current. This variable does not specify where the meeting is to take place.

15. random delay: This variable is used to regulate the routine supply missions. The delay is usually in days and represents the number of days either an FOB, AOB, or SOF Team must operate without being re-supplied.
16. residual cargo: This variable indicates the total amount of supplies a specific aircraft can hold. For the five aircraft types used in the model Chinook CH-47, Black Hawk UH-60, Hercules C-130, Sherpa C-23, and CASA CN-235, the residual cargo values are 22,000, 2,640, 36,000, 3,000, 13,277 respectively.
17. resupply: This variable is a binary variable. When set to ‘1’ an emergency resupply mission is scheduled.
18. rindex: This variable indicates the location requiring emergency resupply. It usually takes the value of a FOB, AOB or SOF Team.supply: This variable is binary and when set to ‘1’ indicates that a routine supply mission is underway.
19. t1-t10: These ten variables are used as indices for the SOF Teams assigned to a particular AOB. Depending on your implementation you may have to increase this number. These variables are used for two functions, (1) to signal that an SOF Team must be visited as part of a contingency mission, and (2) to represent the amount of supplies that needs to be transported to that particular team.
20. tmedivac: This variable indicates that a casualty evacuation mission is underway to one of the SOF Teams. The *mindex* variable is used to specify the particular SOF team.
21. total time: This variable keep track of the total time (hrs) each aircraft is in the system. The variable is reset when it exceeds 24.
22. tresupply: This variable indicates that an emergency resupply mission is underway to one of the SOF Teams. The *rindex* variable is used to specify the particular SOF team.
23. type: The type variable indicates whether the supply mission is a trailet mission or non-trailet mission.
24. wait time: If set this variable indicates the amount of stand-down time an aircraft must immediately undertake. The variable is only set when an aircraft exceeds it total daily flight hours.

REFERENCES

1. *Extend™ simulation software for the next millennium: Extend User’s Manual,* version 4, Imagine That Inc., 6830 Via Del Oro, Suite 230, San Jose, CA 95119-1353, USA, 1997

2. *Extend+Manufacturing™ industrial and commercial modeling, Extend+Manufacturing User’s Manual,* version 4, Imagine That Inc., 6830 Via Del Oro, Suite 230, San Jose, CA 95119-1353, USA, 1997.

3. University of Texas, Perry-Castaneda Library Map Collection, University of Texas at Austin Library Online, 2001.